Computer Games Series

GIOCHIAMO CON VIC 20

GREMESE EDITORE

Computer Games Series

GIOCHIAMO CON VIC 20

GIOCHIAMO CON VIC 20

Alistair Gourlay

GREMESE EDITORE

Computer Games Series
Periodico mensile
N. 2 - Aprile 1984
Registrazione Trib. di Roma N. 138/84
del 24 marzo 1984
Direttore responsabile: Grazia Valci

Titolo originale
Games for your VIC 20

Traduzione dall'inglese
Simonetta Cristiani

Edizione italiana a cura di
Giancarlo Zagarese

Design
Ray Hyden

Illustrazioni
Sue Walliker

Fotocomposizione
Typo-centro s.n.c. - Roma

Stampa
Litopat - Verona

© 1983 Interface/Virgin Books
© 1984 GREMESE EDITORE
Via Virginia Agnelli, 88 - 00151 Roma

Tutti i diritti riservati. Nessuna parte di questo libro può essere riprodotta, registrata o trasmessa, in qualsiasi modo o con qualsiasi mezzo, senza il preventivo consenso dell'Editore.

ISBN 88-7605-117-1

ALASTAIR GOURLAY
L'AUTORE

Alastair Gourlay, che studia attualmente presso il Paisley College, nelle vicinanze di Glasgow, è anche l'autore di *34 Amazing Games for the IK ZX81,* edito in Gran Bretagna.

TIM HARTNELL
CURATORE DELL'EDIZIONE INGLESE

Tim Hartnell è un eminente giornalista la cui esperienza computeristica ha contribuito a determinare il successo della Technical Consumer Press. È anche l'autore di diversi libri, tra i quali: *Getting Acquainted With Your ZX81, Let Your BBC Micro Teach You to Program* e *Programming Your ZX Spectrum.*

GIANCARLO ZAGARESE
CURATORE DELL'EDIZIONE ITALIANA

Giancarlo Zagarese, insegnante di discipline scientifiche, è autore di oltre 250 articoli e di vari volumi nei settori dell'elettronica e dell'attività subacquea. Per GREMESE EDITORE ha già collaborato in *Il Sub per tutti* e *Il Sub in apnea* della serie "gli Abbiccì".

SUE WALLIKER
L'ILLUSTRATRICE

Sue Walliker è un'illustratrice free-lance.

RINGRAZIAMENTI

L'Autore desidera ringraziare Ken Staples, Ric Kube, Mike O'Neil, Adam Burbidge e Kevin Smith per aver contribuito alla stesura materiale di questo libro. L'Autore ringrazia particolarmente John Cade, che ha dedicato molte ore alla sperimentazione, al "debugging" e a far "girare" programmi illustrati in questo volume.

INDICE

Introduzione 13
Introduzione dell'autore 14
Prefazione all'edizione italiana 15
I caratteri di controllo 17
New Kid in Town (Un nuovo ragazzo tra noi) 19
Combat (Battaglia) 21
Trail Blazer (L'esploratore) 24
Piano 2001 (Il pianoforte del 2001) 26
Sketchpad (Il taccuino degli schizzi) 27
Amaze (Sorpresa) 28
Wumpus (I pipistrelli) 30
Digit-Dodge (La caccia al numero) 32
Tail Gunner (Guastatore in retroguardia) 33
Space Dock (L'attracco nello spazio) 35
Golf 37
Breakout (L'evasione) 40
Moonbase (La base lunare) 42
Life (Vita) 44
Fruit Machine 46
Spots from Outer Space (Macchie dallo spazio esterno) 49
Destroyer (L'incrociatore spaziale) 51
Car Race (Una corsa automobilistica) 54
Sub Chase (Inseguimento negli abissi) 56
Reversi/Othello 58
Apollo II 61
Sound Effects (Effetti sonori) 62
Hangman (L'impiccato) 64
Simon 67
Pontoon (Il gioco del "ventuno") 69
Ski Run (Gara di sci) 72
Diamond Glory (Lo splendore del diamante) 73
Zombies 75
Micro Music 79
Rat Trap (La trappola) 81
Jupiter Run (In viaggio verso Giove) 83
Vogon 86
Come scrivere programmi migliori 89
Glossario 97
Traduzioni 113

Introduzione

Il vostro computer è in attesa per sfidarvi. Rapidi «games» di grafica, giochi di concentrazione, di parole e di enigmistica sono tutti qui pronti a farvi divertire.
Nel libro vi sono una notevole varietà di giochi i cui programmi sono stati scritti da alcuni fra i più giovani e abili programmatori che lavorano attualmente in Gran Bretagna. L'esaminare i programmi dei giochi vi dà la possibilità di apprendere raffinate tecniche e sottili metodi di programmazione che voi stessi potete poi applicare. Inoltre, una volta che avete conosciuto a fondo i programmi presentati dal libro, potreste senz'altro provare a migliorarli — un programma non è mai "perfetto" — arricchendo le vostre capacità di programmazione. Adesso voltate pagina e incominciate a "battere" i programmi. Sono certo che il vostro divertimento sarà pari a quello che abbiamo provato durante la stesura di questo volume.

Introduzione dell'autore

Il VIC 20 è un computer particolarmente flessibile, eppure si realizzano molti programmi destinati al VIC 20 senza tener conto di questo fatto. Questo gruppo di programmi è stato scelto proprio per dimostrare, in maniera convincente, come un computere VIC 20 possa essere rapido e flessibile. Dal creare musica alle battaglie spaziali, dal gioco del golf alle battaglie fra carri armati, questi programmi vi offriranno ore di divertimento e vi insegneranno qualcosa in più sulle affascinanti possibilità del vostro computer.
In ogni modo, non vorrei che voi consideraste questi giochi come la Sacra Scrittura e che quindi non vi sia permesso di interferire nelle versioni finali. Piuttosto dovreste usare questi programmi come trampolino di lancio per i vostri giochi, per approfondire la vostra conoscenza nel settore, per scoprire, a poco a poco, l'adattabilità e perfino la compagnia di un VIC 20 quando è pienamente e completamente utilizzato.

Prefazione all'edizione italiana

È con piacere che ho accolto l'invito dell'editore Gremese per curare una serie, finalmente in italiano, di volumetti sui videogiochi. Molti acquirenti di piccoli e medi personal computer, sia che lo utilizzino personalmente sia che ne abbiamo fatto oggetto di regalo ai propri figli si sono fatti trascinare dalla pubblicità che precisava «al prezzo di un semplice video-gioco acquistate un intero computer». È vero ed è stato un buon acquisto. Per imparare ad utilizzare un computer in modo semplice e divertente non c'è però niente di meglio della via ludica. Non giochi comprati e fruiti passivamente, però, ma "creati" e vissuti, istruzione dopo istruzione prima compiendo e poi personalizzando o "inventando", in modo da accedere gradualmente nel nuovo mondo dell'informatica.

g.z.

In ogni "listing" sono stati usati i seguenti caratteri di controllo.

F 1 –
F 2 –
F 3 –
F 4 –
F 5 –
F 6 –
F 7 –
F 8 –

Cursore:

Sopra – Sotto –
A sinistra – A destra –

Pulizia video –
Posizione base del cursore –

Nero –
Bianco –
Rosso –
Azzurro –
Violaceo –
Verde –
Blu –
Giallo –

Nota: nel gioco del "golf" vi sono due "listing" per aiutarti a registrare il testo in "lower case"; in ogni rettangolo vi sono: comandi di tastiera del VIC 20.

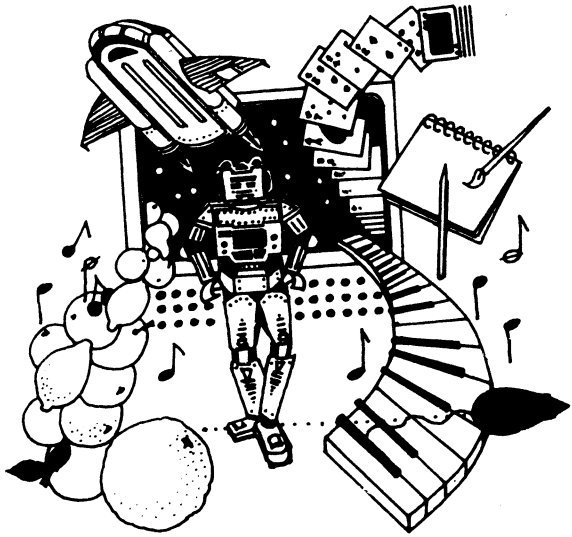
NEW KID IN TOWN

Questo è il programma migliore per mostrare a tutti le ottime caratteristiche del tuo VIC quando sei stato il primo a comprarlo. È stato scritto da Ken Staples e nessuna introduzione è necessaria. Devi solo battere a macchina il programma e poi usarlo per far colpo sui tuoi amici!

```
1 REM NEW KID IN TOWN
10 SL=7680:CL=38400:S1=36874:S2=36875:S3=36876:S4=36877:V=36878
11 DIMD(8)
12 PRINT"■■■■■■ DEMONSTRATION TAPE":PRINT"■■■■ A LIGHT HEARTED":PRINT"■■■ LOOK AT
 A FEW";
13 PRINT"■OF THE":PRINTSPC(5)"CAPABILITIES":PRINTSPC(6)"■■■OF THE VIC":FORI=1TO10
000:NEXT
20 PRINT"■■■■"SPC(6)"■HI THERE!":PRINT"■ MY NAME IS VIC.":PRINT"■"SPC(9)"YOUR
30 PRINT"■ FAITHFUL SERVANT":PRINT"■WOULD YOU LIKE TO SEE ■SOME OF THE WONDERFU
L"
40 PRINT"■THINGS I CAN DO?":PRINT"■"SPC(9)"YES!!!"
50 PRINT"■GOOD!ARE YOU SITTING■ COMFORTABLY?":FORI=1TO10000:NEXT
60 PRINT"■■■■■■■■■■WATCH ME CHANGE BORDER":PRINTSPC(1)"AND SCREEN COLOUR IN"
65 PRINTSPC(2)"■255 DIFFERENT WAYS"
70 FORI=1TO5000:NEXT:FORI=1TO255:POKE36879,I:PRINT"■"I:FORJ=1TO300:NEXTJ,I
75 FORI=1TO5000:NEXT:FORI=1TO255:POKE36879,I:NEXT
80 PRINT"■■■■■■■■■ JUST LET ME CHANGE"
90 PRINT"■TO MY ORIGINAL COLOUR":FORI=1TO5000:NEXT:POKE36879,27
95 FORI=1TO1000:NEXT
100 PRINT"■■■■ NOW LET US PLAY THE PIANO.JUST PRESS THE FOLLOWING KEYS"
110 PRINT"■1155665"SPC(9)"4433221      5544332"
120 PRINT"5544332"SPC(9)"1155665"SPC(9)"443322189"
130 FORI=1TO8:READD(I):NEXT
131 DATA223,227,230,231,234,236,238,239,223,223,234,234
135 DATA236,236,234,231,231,230,230,227,227,223,234,234,231,231
136 DATA230,230,227,234,234,231,231,230,230,227,223,223,234,234,236,236,234,231,
231,230
137 DATA230,227,227,223,239
140 POKEV,15
150 GETA$:IFA$=""THEN150
160 I=VAL(A$):IFI=0ORI=9THENPOKES2,0:GOTO180
170 POKES2,0:FORJ=1TO25:NEXT:POKES2,D(I):GOTO150
180 PRINT"■■■■■■■■ WHY NOT TRY AGAIN?   ■ALTERNATIVELY,WHY     ■NOT TRY TO COMPOS
E"
181 PRINT"■ YOUR OWN TUNE?JUST   ■PRESS'Y' OR 'N'"
185 GETZ$:IFZ$<>"Y"ANDZ$<>"N"THEN185
190 IFZ$="Y"THENRESTORE:GOTO100
250 PRINT"■■■■ LET US SEE IF I CAN  ■REMEMBER THAT TUNE YOU  ■WERE TRYING TO PLA
Y"
260 FORI=1TO4000:NEXT
300 POKEV,15:FORI=1TO43:READD(0):POKES2,D(0):FORJ=1TO400:NEXT:POKES2,0
301 A=250:IFI=7ORI=14ORI=21ORI=28ORI=35THENA=500
```

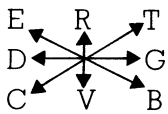
```
305 FORK=1TOA:NEXT
310 NEXTI:POKES2,0:POKEV,0
320 PRINT"▒▒▒▒▒▒▒▒ NOT EXACTLY BACH BUT ▒I'M TAKING LESSONS!!!"
340 FORI=1TO5000:NEXT:PRINT"▒▒▒▒▒▒▒▒ NOW LET US HAVE A▒      FEW SOUND EFFECTS"
345 FORI=1TO5000:NEXT
350 POKES4,220:FORL=15TO0STEP-1:POKEV,L:FORM=1TO300:NEXTM,L:POKES4,0:POKEV,0
360 POKEV,15:POKES4,0:FORO=248TO253:POKES4,0:FORI=1TO100:NEXTI,O
370 FORO=253TO248STEP-1:POKES4,0:FORI=1TO100:NEXTI,O:POKES4,0:POKEV,0
380 POKEV,3:FORL=230TO128STEP-1:POKES3,L:FORM=1TO20:NEXTM,L:POKES3,0:POKES4,200
390 FORL=15TO0STEP-.05:POKEV,L:NEXT:POKES4,0
400 POKEV,15:FORL=1TO10:POKES2,200:FORM=1TO500:NEXT:POKES2,0:POKES3,200:FORM=1TO
500:NEXT
410 POKES3,0:NEXT:POKEV,0
420 POKES4,135:FORL=1TO10:D=INT(RND(1)*90)+30:FORM=0TO10:POKEV,M:FORN=1TOD:NEXTN
,M
430 FORM=15TO0STEP-1:POKEV,M:FORN=1TOD:NEXTN,M,L:POKEV,0:POKES4,0
440 POKEV,15:FORL=1TO15:FORM=200TO220+L*2:POKES3,M:NEXTM,L:POKEV,0:POKES3,0
450 POKEV,15:B=0:FORL=128TO255STEP11:POKES1,L:FORM=1TO10:NEXT:B=B+1
460 IFB=3THENB=0:POKES1,0
470 NEXTL:POKEV,0:POKES1,0
480 PRINT"▒NOW FOR A VARIATION OF▒KNOCKING DOWN THE WALL ▒GAME"
490 PRINT"▒ OR WHY NOT USE IT TO    ▒DESIGN YOUR NEXT     ▒LOUNGE CARPET?!!"
495 FORI=1TO10000:NEXT
500 PRINT"▒":POKE36879,9:POKEV,15
510 FORI=1TO500:POKE7679+INT(RND(1)*506)+1,102:NEXT
520 X=1:Y=1:DX=1:DY=1
530 POKE7679+X+22*Y,81:FORI=1TO10:NEXT:POKE7679+X+22*Y,32
540 X=X+DX:IFX=0ORX=21THENDX=-DX:POKES3,220
550 Y=Y+DY:IFY=0ORY=22THENDY=-DY:POKES3,230
560 POKES3,0:IFPEEK(7679+X+22*Y)=102THENDX=-DX:POKES3,180
565 A=A+1:IFA=500THENPOKES3,0:POKEV,0:POKE36879,27:RUN
567 B=B+1:IFB=15THENB1=B1+1:B=0
568 IFB1=16THENB1=0
570 POKE36879,B1:GOTO530

READY.
```



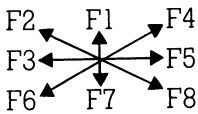
COMBAT

In questo gioco a due, scritto da Ric Kube, dell'Australia del Sud, carri armati si combattono fino alla morte. Entrambi i carri armati sono ben protetti e possono subire fino a dieci colpi prima di essere distrutti. Mine mortali, visualizzate sotto forma di rombi, sono sparse attorno al campo di battaglia. Se un carro armato urta contro una mina, perderà una delle sue dieci «vite». Ci sono altri ostacoli, visualizzati come quadrati colorati che, dopo essere stati colpiti una volta, diventano quadrati a puntini. Se il carro armato colpisce un quadrato a puntini, sarà totalmente distrutto. Il giocatore che per primo perderà le sue dieci vite perderà anche la battaglia.
I caratteri di controllo per il giocatore di sinistra sono:



F per sparare

I caratteri di controllo per il giocatore di destra sono:



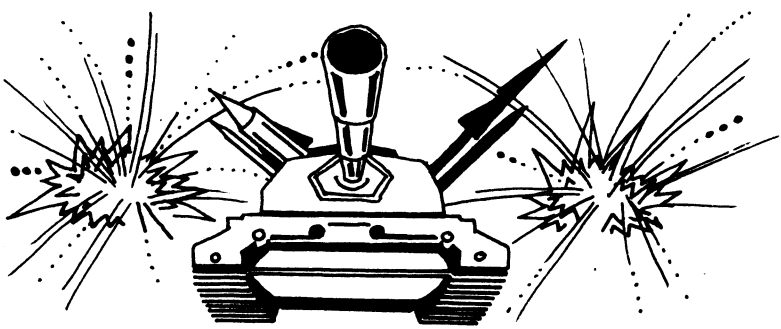
Usa la «freccia in alto» per sparare

Il punteggio da raggiungere è di dieci punti. Si segna un punto sia sparando verso il carro armato dell'avversario sia quando il tuo avversario incontra una mina. I missili voleranno sulle mine e si distruggeranno colpendo le pareti esterne.

```
100 REM**COMBAT
110 REM**RIC KUBE
120 GOSUB850
130 GOSUB840:PRINT"□"
140 POKE36879,93
150 FORX=1TO20
160 POKE7680+X,100:POKE8098+X,99
170 NEXT
180 FORY=1TO18
190 POKE7680+Y*22,103:POKE7701+Y*22,101
```

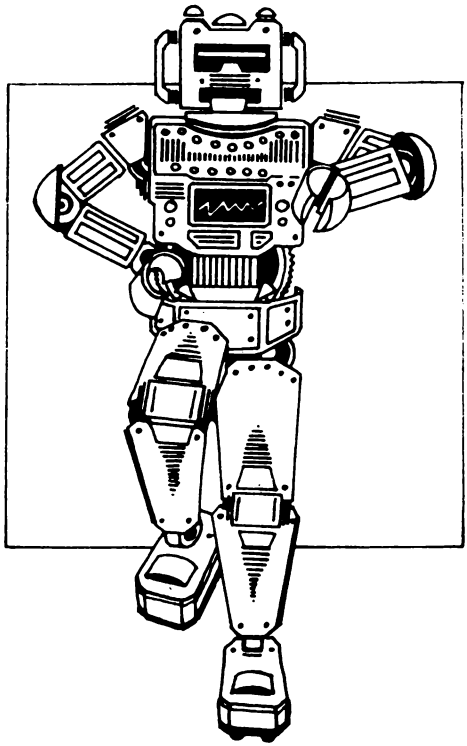
```
200 NEXTY
210 FORY=1TO18
220 FORX=1TO20
230 R=RND(1)*10
240 IFR>9.3ANDR<9.75THENPOKE7680+22*Y+X,160
250 IFR>9.75THENPOKE7680+22*Y+X,90
260 NEXT:NEXT
270 LV=10:LH=3:RV=10:RH=36
280 M=32
290 POKE7903,81:POKE7936,87
300 POKE8126,SL+48:POKE8133,SR+48
310 IFSL=5ORSR=5THENPOKE36879,91
320 IFSL=9ORSR=9THENPOKE36879,90
330 IFSL=10ORSR=10THEN740
340 A$=""
350 GETA$:IFA$=""THEN350
360 IFA$="C"ORA$="■"THENR=-1:D=1
370 IFA$="V"ORA$="■"THENR=0:D=1
380 IFA$="B"ORA$="■"THENR=1:D=1
390 IFA$="D"ORA$="■"THENR=-1:D=0
400 IFA$="F"ORA$="↑"THEN570
410 IFA$="G"ORA$="■"THENR=1:D=0
420 IFA$="E"ORA$="■"THENR=-1:D=-1
430 IFA$="R"ORA$="■"THENR=0:D=-1
440 IFA$="T"ORA$="■"THENR=1:D=-1
450 IFR=0ANDD=0THEN340
460 IFASC(A$)>130THENRR=R:RD=D:V=RV:H=RH:P=87:S=SL:M=RM
470 IFASC(A$)<130THENLR=R:LD=D:V=LV:H=LH:P=81:S=SR:M=LM
480 C=7680+H+22*V
490 N=7680+H+22*(V+D)+R
500 IFPEEK(N)>80ANDPEEK(N)<>90THEN300
510 IFPEEK(N)=32THENPOKEC,M:M=32:POKEN,P
520 IFPEEK(N)=90THENPOKEC,M:M=90:POKEN,P:GOSUB680:S=S+1
530 H=H+R:V=V+D
540 IFASC(A$)>130THENRV=V:RH=H:SL=S:RM=M
550 IFASC(A$)<130THENLV=V:LH=H:SR=S:LM=M
560 GOTO300
570 IFA$="↑"THENV=RV:H=RH:R=RR:D=RD:Q=87
580 IFA$="F"THENV=LV:H=LH:R=LR:D=LD:Q=81
590 C=7680+H+22*V
600 N=7680+H+R+22*(V+D)
610 IFPEEK(N)=32THENPOKEC,Q:POKEN,46:H=H+R:V=V+D:Q=32:GOTO590
620 IFPEEK(N)=102THENPOKEC,Q:GOSUB680:POKEN,32:GOTO300
630 IFPEEK(N)=160THENPOKEC,Q:GOSUB680:POKEN,102:GOTO300
640 IFPEEK(N)=90THENPOKEC,Q:POKEN,46:H=H+R:V=V+D:Q=90:GOTO590
650 IFPEEK(N)=81THENSR=SR+1:POKEC,Q:GOSUB680:GOTO300
660 IFPEEK(N)=87THENSL=SL+1:POKEC,Q:GOSUB680:GOTO300
670 IFPEEK(N)>98THENPOKEC,Q:GOTO300
680 Z1=PEEK(N+1):Z2=PEEK(N-1):Z3=PEEK(N+22):Z4=PEEK(N-22)
690 POKEN+1,42:POKEN-1,42:POKEN+22,42:POKEN-22,42
700 FORX=0TO50:NEXT
710 POKEN+1,Z1:POKEN-1,Z2:POKEN+22,Z3:POKEN-22,Z4
720 POKE36878,15:POKE36877,255:FORB=1TO50:NEXT:POKE36878,0
730 RETURN
740 IFSL=10THENA$="LEFT"
750 IFSR=10THENA$="RIGHT"
760 POKE36879,93
770 PRINT"■■■ THE GAME WAS WON BY":PRINT"   THE "A$" PLAYER"
780 PRINT"ANOTHER GAME?(Y/N)"
790 GETN$:IFN$=""THEN790
```

```
800 IFN$="Y"THEN830
810 IFN$="N"THENEND
820 GOTO790
830 GOSUB840:GOTO130
840 A$="":R=0:D=0:SL=0:SR=0:LM=32:RM=32:RETURN
850 PRINT"COMBAT"
860 PRINT"THE OBJECT OF THE GAME IS TO SCORE 10 POINTS"
870 PRINT"YOU MAY SCORE A POINT IN TWO WAYS-"
880 PRINT"1)SHOOTING YOUR OPPON-ENT'S TANK":PRINT"2)YOUR OPPONENT MOVINGON TO A
MINE"
890 PRINT"MISSILES WILL FLY OVERMINES AND DISINTEGRATE ON HITTINGTHE OUTER WALL
"
900 PRINT"EACH PLAYER HAS 9     CONTROLS.":PRINT"  PRESS ANY KEY TO        CONT
INUE"
910 GETA$:IFA$=""THEN910
920 PRINT"LEFT PLAYER"
930 PRINT"  E R T"
940 PRINT"  D F G"
950 PRINT"  C V B"
960 PRINT"RIGHT PLAYER"
970 PRINT"F2 F1 F4"
980 PRINT"F3  ↑ F5"
990 PRINT"F6 F7 F8"
1000 PRINT"PRESS ANY KEY TO CONTINUE"
1010 GETA$:IFA$=""THEN1010
1020 PRINT"TO MOVE PRESS THE KEY FOR THE DIRECTION YOU WANT TO GO"
1030 PRINT"TO FIRE PRESS THE        'F' OR '↑' KEYS"
1040 PRINT"THE MISSILE FIRES IN THE DIRECTION YOU LAST MOVED"
1050 PRINT"LEFT TANK-'●'
1060 PRINT"RIGHT TANK-'o'
1070 PRINT"PRESS ANY KEY TO BEGIN"
1080 GETA$:IFA$=""THEN1080
1090 RETURN
```



TRAIL BLAZER

Sei intrappolato in un'arena, la zona piena di puntini, con un robot controllato dal computer. Quando tu e il robot vi spostate, lasciate dietro di voi delle tracce. Se urti contro le mura dell'arena o incroci le tracce, sarai fulminato.
Questi sono i punteggi:
Se intrappoli il robot (il verde O) = 200 punti
Se colpisci il punto rosso porpora = 50 punti
Se colpisci il puntino bianco = 10 punti.
La velocità del gioco aumenta ad ogni partita.



```
10 RESTORE
13 GOSUB3000:GOSUB2000
15 X=RND(-TI):DEFFNR(X)=(PEEK(7663+X)=0)
20 D(0)=1:D(1)=22:D(2)=-1:D(3)=-22
30 L=150+(KJ$="J")*50
40 PP=46:PC=437:DP=1:DC=2:SC=7680:CL=38400:POKE36878,15
45 GOSUB1000
50 IFKJ$="J"THEN62
55 GETA$:POKE198,0
60 ND=22*((A$="P")-(A$=".."))+(A$="L")-(A$=":"):IFND<>0THENDP=ND
61 GOTO70
62 SYS7600:X1=0:FORX=0TO3:X1=X1-FNR(X):NEXT:IFX1>1THEN70
65 ND=22*(FNR(0)-FNR(1))+FNR(2)-FNR(3):IFND<>0THENDP=ND
70 NP=PP+DP:X1=PEEK(SC+NP):IFX1<>46THEN700
75 POKE36874,128:S=S+10
80 POKESC+PP,160:POKECL+PP,2:POKESC+NP,42:POKECL+NP,2:PP=NP
85 PRINT"[redacted] SCORE "S"[redacted]";:POKE36874,0:POKE36876,0
90 NC=PC+D(DC):IFPEEK(SC+NC)<>46ORRND(1)<.15THEN200
100 POKESC+PC,160:POKECL+PC,5:POKESC+NC,15:POKECL+NC,5:PC=NC
110 FORX=0TOL:NEXT:GOTO50
200 FORX=1TO19:DC=INT(RND(1)*4):IFPEEK(SC+PC+D(DC))=46THENNC=PC+D(DC):GOTO100
210 NEXT
220 S=S+200:L=L/2
230 FORX=1TO4:FORY=128TO160:POKE36875,Y:NEXT:NEXT:POKE36875,0:POKE198,0:GOTO40
700 IFX1=81THENPOKE36876,200:S=S+50:GOTO80
800 POKE36877,250:FORX=1TO2000:NEXT:POKE36877,0
810 PRINT"[redacted]     GAME   OVER       ":FORX=1TO5000:NEXT:POKE198,0:S=0:G
OTO10
1000 PRINT"[redacted]";:FORX=1TO20:PRINT"[redacted]";:NEXT
1010 PRINT"[redacted]";
1020 FORX=1TO5:X1=INT(RND(1)*20)+1+(INT(RND(1)*20)+1)*22
1030 POKE7680+X1,160:POKE38400+X1,0:NEXT
1040 FORX=1TO5:X1=INT(RND(1)*20)+1+(INT(RND(1)*20)+1)*22
1050 POKE7680+X1,81:POKE38400+X1,4:NEXT:POKESC+46,46:POKESC+437,46
1060 POKECL+46,1:POKECL+437,1:RETURN
2000 POKE36879,25
2010 PRINT"[redacted]    TRAIL BLAZER"
2020 PRINT"[redacted]  MOVE AROUND THE    SCREEN USING THE      JOYSTICK OR-"
2030 PRINT"          [redacted]"
2040 PRINT"[redacted]SCORING-":PRINT".-10 PTS":PRINT"[redacted]-50 PTS":PRINT"[redacted]TRAP-200 PTS"
2960 PRINT"[redacted]    PRESS ANY KEY"
2970 GETA$:IFA$=""GOTO2970
2980 POKE36879,110:IFKJ$<>""THENRETURN
2990 PRINT"[redacted]KEYBOARD OR JOYSTICK?":PRINT"[redacted]   (K/J)"
2991 GETKJ$:IFKJ$<>"K"ANDKJ$<>"J"THEN2991
2992 RETURN
3000 POKE51,176:POKE52,29:POKE55,176:POKE56,29
3010 FORX=7600TO7661:READA:POKEX,A:NEXT
3020 RETURN
3030 DATA169,0,141,19,145,169,127,141,34,145,173,17,145,141,238,29,169,4,45,238,
29
3040 DATA141,239,29,169,8,45,238,29,141,240,29,169,16,45,238,29,141,241,29,169,3
2
3050 DATA45,238,29,141,243,29,173,32,145,41,128,141,242,29,169,255,141,34,145,96

READY.
```

PIANO 2001

Questo programma trasforma il VIC in un magico, futuristico pianoforte. Il pianoforte utilizza i generatori di note medie e alte del VIC, permettendoti cosí di suonare due note alla volta. La tastiera è usata nel modo seguente:

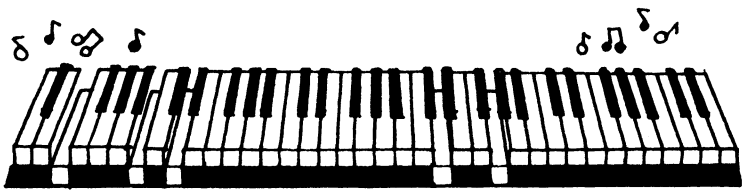
Note Alte:
Q W E R T Y U I — F3 — off

Note medie:
A S D F G H J K — F5 — off

F1 è usato per effettuare il movimento tra le ottave basse e le ottave medie. In questo modo viene aumentata l'estensione della tastiera del pianoforte.

```
1 FORX=36875TO36877:POKEX,127:NEXT
3 POKE36879,8:PRINT"[illegible]"
5 DIMNO(2,9)
6 OC=1
10 FORX=1TO9:READNO(1,X):NEXT
15 FORX=1TO9:READNO(2,X):NEXT
100 DATA128,141,153,159,170,179,187,191,127,191,198,204,207,213,217,221,223,127
200 GETA$
205 IFA$="[illegible]"THENOC=OC+1:IFOC=3THENOC=1
220 B$="ASDFGHJK[illegible]":GOSUB1000:IFX<>-1THENT=36875:GOSUB2000
230 B$="QWERTYUI[illegible]":GOSUB1000:IFX<>-1THENT=36876:GOSUB2000
235 FORX=15TO6STEP-3:POKE36878,X:NEXT
236 FORX=6TO15STEP3:POKE36878,X:NEXT
240 GOTO200
1000 FORX=1TOLEN(B$):IFA$=MID$(B$,X,1)THENX=NO(OC,X):RETURN
1010 NEXT:X=-1:RETURN
2000 IFX=PEEK(T)THENPOKET,X-10
2010 FORN=PEEK(T)TOXSTEPSGN(X-PEEK(T))*ABS(X-PEEK(T))/99:POKET,N:NEXT:POKET,X:RE
TURN

READY.
```

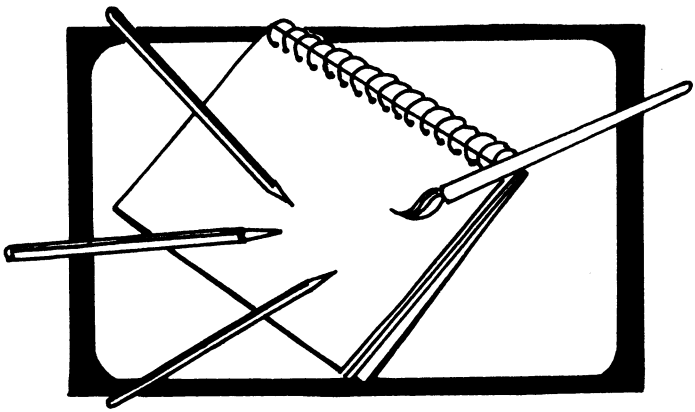


SKETCHPAD

Lascia sfogare la tua immaginazione con questo coloratissimo programma che si basa su disegni di schizzi. Usa i seguenti tasti: "P" = sopra, "." = sotto, "L" = sinistra, ":" = destra, per muovere il cursore (il quadrato a scacchiera) sul video. Premi uno dei due tasti di "shift" per disegnare, o il tasto di "shift lock" quando vuoi muovere il cursore. Se il tasto delle maiuscole viene premuto, il cursore cancellerà tutto appena si muove.
Per cancellare l'immagine dallo schermo, devi solo tenere abbassato il tasto "." finché il disegno si sposti verso l'alto del video. Se vuoi cambiare il colore per il tuo disegno, premi i tasti da uno a otto, a seconda del colore desiderato. Attenzione: Il tasto delle maiuscole = shift e il tasto di "arresto" = shift-look delle maiuscole non dovrebbero essere premuti mentre si cambia il colore.

```
1 REM SKETCHPAD
10 POKE36879,136:PRINT"[illegible]";:CL=6:CL$="[illegible]":POKE36878,15
20 X=PEEK(197)
25 A$=" "
30 IFPEEK(653)<>0THENA$="[illegible]"
40 IFX=13THENPRINTA$;"[illegible]";
50 IFX=37THENPRINTA$;"[illegible]";
60 IFX=21THENPRINTA$;"[illegible]";
70 IFX=45THENPRINTA$;
75 GETA$:A=VAL(A$):IFA<>0THENCL=A
80 PRINT"[illegible]";MID$(CL$,CL,1);"[illegible]";:POKE36875,160+CL*10
90 FORX=1TO200:NEXT:PRINT" [illegible]";:POKE36875,0:GOTO20

READY.
```



AMAZE

Questo è un gioco semplice, ma affascinante, scritto da Mike O'Neill, basato su un sistema di labirinti, nel quale devi usare tutta la tua bravura contro il computer. Il cerchio è sotto il tuo controllo, il rombo è sotto il controllo del computer. I livelli di gioco sono dieci, a diverse velocità, e ci sono tre diversi labirinti. Il tuo tempo è visualizzato in fondo al video da una barra orizzontale che diventa sempre più piccola a mano a mano che i secondi scorrono. I tasti usati sono "A" = sopra, "S" = sotto, "K" = sinistra, "L" = destra. I labirinti possono essere cambiati alterando i sottoprogrammi tra le linee 5000 e 7010. Se si inserisce l'espansione di memoria del computer, possono essere aggiunti altri labirinti. I tasti usati possono essere cambiati alterando le linee fra la 350 e la 380.

```
10 PRINT"[illegible]        AMAZE  ":PRINT"[illegible]  BY MIKE O'NEILL[illegible]":POKEV,0
20 R=RND(-TI):S=36876:V=36878:PRINT
30 INPUT" ENTER LEVEL (0-9)";R:IFR>9ORR<0THEN30
40 IR=200-R*20
50 PRINT
60 INPUT" WHICH MAZE (1-3)";I%
70 IFI%<1ORI%>3THEN60
80 POKE36879,57:PRINT"[illegible]"
90 ONI%GOSUB7000,5000,6000
100 PRINT"[illegible]                    [illegible]"
140 X1=INT(RND(1)*20)
150 X2=INT(RND(1)*20):IFABS(X1-X2)<6THEN150
160 Y1=INT(RND(1)*18)
170 Y2=INT(RND(1)*18):IFABS(Y1-Y2)<6THEN170
180 P1=7702+X1+Y1*22:IFPEEK(P1)<>32THEN140
190 P2=7702+X2+Y2*22:IFPEEK(P2)<>32THEN150
200 POKEP1+30720,1:FORJ=1TO10
210 POKEP1,32:FORI=1TO50:NEXT
220 POKEP1,81
230 POKES,202:POKEV,15:FORI=1TO50:NEXT:POKEV,0:NEXT
250 POKEP2+30720,2:FORJ=1TO10
260 POKEP2,32:FORI=1TO50:NEXT
270 POKEP2,90
280 POKES,151:POKEV,15:FORI=1TO50:NEXT:POKEV,0:NEXT
290 I%=0
300 I%=I%+1:GOSUB4000:A$="":IFI%>100THEN8000
310 POKES,150+I%:POKEV,15
320 GETB$:IFB$=""THEN340
330 A$=B$:GOTO320
340 IFA$=""THEN400
350 IFA$="P"THEND1=-22
360 IFA$="."THEND1=22
370 IFA$="L"THEND1=-1
380 IFA$=";"THEND1=1
400 POKEP1,32:P1=P1+D1:IFPEEK(P1)=90THEN1000
```

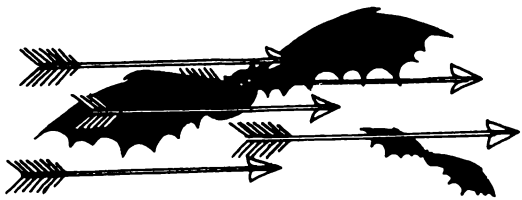
```
410 IFPEEK(P1)<>32THENP1=P1-D1:D1=0
420 POKEP1+30720,1:POKEP1,81
500 GOSUB2000:POKEP2,32:P2=P2+D2:IFPEEK(P2)=81THEN1000
520 POKEP2+30720,2:POKEP2,90
600 POKEV,0:FORI=1TOIR:NEXT:GOTO300
1000 GOSUB3000:SS=100-I%:IFSS<0THENSS=0
1010 PRINT"■  ■■    SCORE";SS
1020 INPUT"■■■■■■■ANOTHER GAME (Y/N)";A$
1030 IFA$="Y"THENCLR:GOTO20
1040 IFA$="N"THENPOKE36879,27:END
1050 GOTO1020
2000 N=0
2010 IFPEEK(P2+1)=32ORPEEK(P2+1)=81THENN=N+1:D(N)=1
2020 IFPEEK(P2-22)=32ORPEEK(P2-22)=81THENN=N+1:D(N)=-22
2030 IFPEEK(P2-1)=32ORPEEK(P2-1)=81THENN=N+1:D(N)=-1
2040 IFPEEK(P2+22)=32ORPEEK(P2+22)=81THENN=N+1:D(N)=22
2100 R=INT(RND(1)*N+1)
2110 IFN>1ANDD(R)=-D2THEN2100
2120 D2=D(R)
2130 RETURN
3000 POKEP1,42:FORJ=I%TO100:POKES,150+J:FORI=1TO10:NEXTI,J
3005 POKES,0:POKEV,0:POKE36877,220:FORJ=15TO1STEP-1
3010 POKEP1,32:FORI=1TO50:NEXT:POKEV,J
3020 POKEP1,42:FORI=1TO50:NEXT
3030 NEXT:POKE36877,0:POKEV,0:RETURN
4000 IFI%>100THEN8000
4010 J%=I%-INT(I%/5)*5:K%=8163-I%/5
4020 IFJ%=0THENPOKEK%,32
4030 IFJ%=1THENPOKEK%,231
4040 IFJ%=2THENPOKEK%,234
4050 IFJ%=3THENPOKEK%,97
4060 IFJ%=4THENPOKEK%,116
4100 RETURN
5000 PRINT"■└──────────────────┘                    │ │  ┌────── ──┐──────┐  │
 │"
5001 PRINT"■│ │         │     │ │ │└│ │┌──────┴──┐ │ │ │└│   │           │ │ │ │"
5002 PRINT"■│ │ │┌──── ─┐  │ │ │└│ │ │ │       │ │ │ │├─│ │ │   ┌ ──┐ │   │ │"
5003 PRINT"■│ │   │ │       │ │   │└│ │ │ │└──────┘ │ │ │└│ │├─│ │         │ └─┘ │"
5004 PRINT"■│   ├─┤ ┌── ──┘       │└│ │   │ │         ┌─┐ │└│ │ │ │└──── ─┐ │ │ │"
5005 PRINT"■│├─┤ │ │         │ │ │ │└│ └─┤   ┌ ──────┘ │ │└│   │ │ │         │ ├─┤"
5006 PRINT"■│├─┘   ├── ┐ ┌─┤   │└│     │     │       │ │└┴──┴──┴──┴──┴──────┴"
5010 RETURN
6000 PRINT"■┌──┬──────┬──────┐│                    │├─┼─┬─┴──┬──┴─┬─┐ ┌─┤"
6001 PRINT"■│    │     │     │ │ │└│ │┌─┴──────┴─┐ │ │ │└│   │           │ │ │ │"
6002 PRINT"■│   ┌ ┬─┬ ┐   │   │└│ │ │ │   │   │ │ │ │├─┤│   │   ┌─┬─┬─┤   │ │"
6003 PRINT"■│   │ │     │   │ │ │└│ │ │ │└──────┴─┤ │ │└│ │└│   │           └─┘ │"
6004 PRINT"■│   ├─┤ └── ─┴─┘       │└│ │   │       ┌─┐ │└│├─│ │ │ ┌──── ─┐   │ │"
6005 PRINT"■│   │ │ │         │   │└│ └─┘ │ └──────┴─┤ │└│   │         │   │ ├─┤"
6006 PRINT"■│ ┌ ┐   ┌── ┐ ┌─┤   │└│     │     │       │ │└┴──┴──┴──┴──┴──────┴"
6010 RETURN
7000 PRINT"■┌──────────────────┐│                    │└ ┌──── ── ── ──┐  │"
7001 PRINT"■│ │                 │ │└│ │┌─ ─── ── ─┐   │└│   │           │ │ │"
7002 PRINT"■│ │ ┌── ──┐   │     │ │└│ │ │ │       │ │ │ │└│   │ │ ┌─┐ │ │ │ │"
7003 PRINT"■│ │ │ │   │   │   │ │ │└│ │   │   ─   │ │ │ │└│ │ │ │ │   │ │ │   │"
7004 PRINT"■│ │ │ └─┘ │   │   │ │ │└│ │ │ │       │ │ │ │└│   └── ─ ──┘ │ │ │"
7005 PRINT"■│   │           │ │ │└│ └── ─── ──────┘   │└│                 │ │"
7006 PRINT"■│ └── ──── ──────┘ │└                    │└──────────────────┘"
7010 RETURN
8000 PRINT"■■■■■■■
```

WUMPUS

Sei intrappolato in un sistema di caverne abitate da pipistrelli. Ci sono due fetidi pipistrelli con te. Sparsi qua e là per le caverne ci sono molti pozzi senza fondo. Se cadi in uno di questi sarai definitivamente perduto. Il tuo solo aiuto sono cinque frecce magiche che possono essere usate per uccidere un pipistrello, ma soltanto quando gli sei abbastanza vicino. Il tuo fidato VIC ti avvertirà quando un pipistrello verrà nei paraggi. Il VIC è il solo alleato che hai nelle caverne. Il tuo computer ti avviserà anche quando ti troverai troppo vicino a uno qualsiasi dei pozzi senza fondo. Il sistema delle caverne è numerato nel modo seguente:

```
    92    94    96    98    100
81    83    85    87    89
    72    74    76    78    80
61    63    65    67    69
    52    54    56    58    60
41    43    45    47    49
    32    34    36    38    40
21    23    25    27    29
    12    14    16    18    20
 1     3     5     7     9
```

Batti il numero della caverna nella quale desideri entrare. Poi spostarti in una caverna in alto, in basso, a destra o a sinistra. C'è qualche altro problema. Non devi impiegare più di 20 minuti a trovare e uccidere i due pipistrelli e ti troverai veramente nei guai se esaurisci tutte le frecce. Ci sono anche delle caverne abitate da vampiri che possono portarti in qualsiasi altra caverna del sistema. Avvertimento finale: non è consigliabile disturbare i pipistrelli, possono diventare molto pericolosi.



WUMPUS

```
1 REM WUMPUS
100 POKE36879,127:PRINT"■"
110 DEFFNR(X)=INT(RND(1)*X)
120 P=1:MA=5
130 NP=FNR(5)+3:FORN=1TONP:PT(N)=FNR(99)+2:NEXT
140 W1=FNR(99)+2:W2=FNR(99)+2:IFW1=W2THEN140
150 TI$="000000"
160 TM=VAL(MID$(TI$,3,2))
170 PRINT"■■CAVE"P"  ■TIME"TM"MIN."
180 PRINT"■■MAGIC ARROWS:"MA
190 IFABS(W1-P)<>1ANDABS(W1-P)<>10ANDABS(W2-P)<>1ANDABS(W2-P)<>10THEN370
200 PRINT"■■■   THERE IS A WUMPUS        NEARBY          ■"
210 PRINT"■FIRE AN ARROW?(Y/N)":POKE198,0
220 GETA$:IFA$<>"Y"ANDA$<>"N"THEN220
230 IFA$="N"THEN470
240 INPUT"WHERE TO";CN
250 MA=MA-1
260 IFABS(CN-P)<>1ANDABS(CN-P)<>10THENPRINT"CAN'T SHOOT HERE":GOTO240
270 IFW1<>CNANDW2<>CNTHENPRINT"■■    YOU MISSED IT     ■":GOTO470
280 IFW1=CNTHENW1=-100
290 IFW2=CNTHENW2=-100
300 PRINT"■■YOU KILLED A WUMPUS■":FORN=1TO1000:NEXT
310 IFW1<>-100ORW2<>-100THEN470
320 POKE36879,25:PRINT"■■■■■■■■■■■YOU HAVE SURVIVED WITH"20-TM"MINUTES LEFT"
330 PRINT"■■■ANOTHER GAME(Y/N)":POKE198,0
340 GETA$:IFA$<>"Y"ANDA$<>"N"THEN340
350 IFA$="N"THENPOKE36879,27:PRINT"■■";:POKE198,0:END
360 RUN
370 FORN=1TONP
380 IFABS(PT(N)-P)<>1ANDABS(PT(N)-P)<>10THEN400
390 PRINT"■■■IT'S DRAUGHTY HERE-       PITS NEARBY!!!    ■":GOTO470
400 NEXT
410 IFFNR(30)>0THEN470
420 PRINT"■■■SUPERBATS HAVE CARRIED    YOU AWAY!!!!!     ■":FORN=1TO3000:NEXT
430 CN=FNR(100)+1:IFCN=PTHEN430
440 P=CN:PRINT"■THEY DROPPED YOU IN       CAVE"P:FORN=1TO3000:NEXT
450 GOTO510
460 IFMA=0THEN620
470 IFMA=0THEN620
480 INPUT"■■WHERE TO NOW";CN
490 IFCN<1ORCN>100ORABS(CN-P)<>1ANDABS(CN-P)<>10THENPRINT"CAN'T GO THERE!!":GOTO
470
500 P=CN
510 FORN=1TONP:IFP=PT(N)THEN560
520 NEXT
530 IFTM=20THEN600
540 IFW1=PORW2=PTHEN580
550 GOTO160
560 POKE36879,8:PRINT"■■■■■■■■■■■■■YOU FELL DOWN A PIT"
570 GOTO330
580 POKE36879,42:PRINT"■YOU WERE EATEN BY A   ■      WUMPUS!!!!!!!!!"
590 GOTO330
600 POKE36879,110:PRINT"■■■■■■■■■■■■ YOU RAN OUT OF TIME  ■"
610 GOTO330
620 POKE36879,76:PRINT"■■■■■■■■■■YOU HAVE NO MAGIC            ARROWS         ■":
GOTO330

READY.
```

DIGIT-DODGE

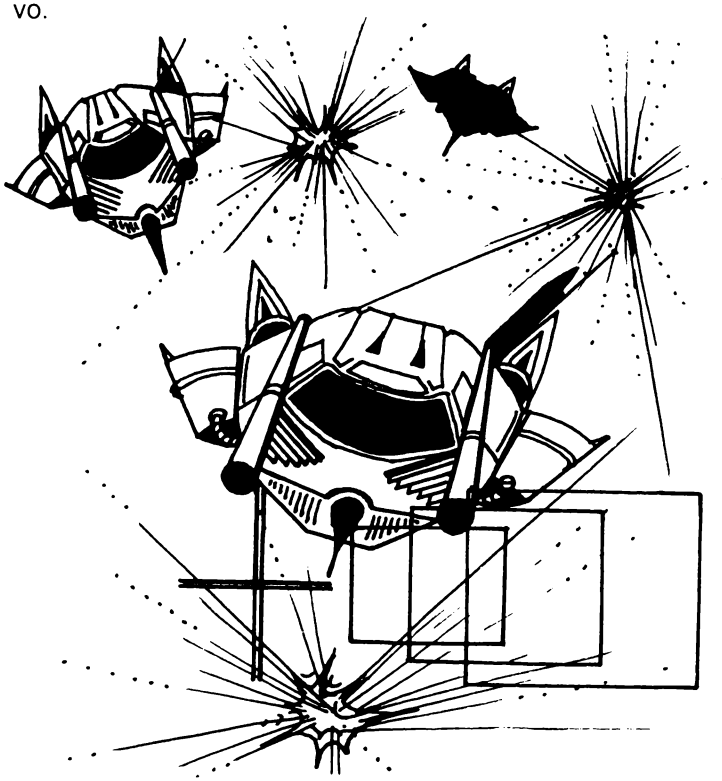
In questo gioco di grafica devi manovrare il tuo brevettato "Digit-dodger" sul video e distruggere i numeri esposti nel giusto ordine, incominciando da 1. Ogni volta che hai completato correttamente ciascun set di numeri, ti sarà dato un nuovo livello di gioco, con l'aggiunta di un numero. Il gioco finisce quando non riesci a completare un livello entro il tempo limite di 30 secondi.
Il gioco ha fine anche quando ti scontri con gli ostacoli, oppure se colpisci un numero che non fa parte della seguenza ordinata per tre volte in una partita. Per muovere il tuo "Digit-Dodger" usa i tasti "P" = sopra, "." = sotto, "L" = sinistra, ":" = destra.

```
100 POKE36879,31:PRINT"";
110 S=2:SC=0:POKE36878,15
120 DEFFNR(X)=INT(RND(1)*X):DEFFNX(X)=X-INT(X/22)*22
130 PRINT"":FORN=1TO20+S*20:X=FNR(462)+44:POKEX+7680,160:POKE38400+X,0:NEXT
140 FORN=1TOS
150 X=FNR(462)+44:IFPEEK(X+7680)<>32THEN150
160 GOSUB440
170 POKEX+7680,N+176:POKEX+38400,N+1:NEXT
180 CP=22:TI$="000000":CS=1:CR=0
190 POKE36874,130
200 NP=CP+22*((PEEK(197)=13)-(PEEK(197)=37))+(PEEK(197)=21)-(PEEK(197)=45)
210 POKE36874,0
220 IFNP=CPORNP<22ORNP>505ORABS(FNX(NP)-FNX(CP))>1THENNP=CP:GOTO280
230 X=PEEK(NP+7680)
240 IFX=CS+176THENGOSUB440:SC=SC+(40-VAL(TI$))*CS:CS=CS+1:GOTO270
250 IFX<>CS+176ANDX<>32THENGOSUB450:NP=CP:CR=CR+1:IFCR=3THEN370
260 PRINT"TIME:"RIGHT$(TI$,2)" SCORE"SC"
270 IFCS>STHEN400
280 POKENP+38400,4:POKE7680+CP,32:POKE7680+NP,81:CP=NP
290 IFTI$<"000030"THEN190
300 FORN=1TO5:GOSUB450:NEXT
310 PRINT" TIME UP "
320 PRINT" ANOTHER GAME(Y/N)  "
330 POKE198,0
340 GETA$:IFA$<>"Y"ANDA$<>"N"THEN340
350 IFA$="N"THENPOKE36879,27:PRINT"";:END
360 RUN
370 PRINT" TOO MANY CRASHES "
380 PRINT" GAME OVER "
390 GOTO320
400 FORN=128TO240:POKE36874,N:POKE36875,N:NEXT:POKE36874,0:POKE36875,0
410 CS=1:S=S+1:IFS>5THENS=5
420 GOTO130
430 GOTO430
440 FORN1=22TO128STEP-1:POKE36876,N1:NEXT:POKE36876,0:RETURN
450 FORN1=200TO140STEP-1:POKE36877,N1:NEXT:POKE36877,0:RETURN

READY.
```

TAIL GUNNER

Tu sei in retroguardia, su un caccia stellare della Federazione che in questo momento è inseguito da una flotta di ricognitori-spia nemici. Devono essere distrutti appena entrano nel tuo campo visivo. Ogni velivolo deve essere abbattuto il più presto possibile per segnare il massimo punteggio. Se eccedi nel tempo limite, esposto sulla parte superiore del video, il nemico potrà informare la sua base e tu perderai il gioco.
Per spostare il tuo campo visivo verso l'alieno, usa gli stessi tasti che hai utilizzato nel "Digit-Dodge" ("P" = sopra "." = sotto, "L" = sinistra, ":" destra). Usa gli stessi tasti per sparare quando l'alieno è al centro del tuo campo visivo.



```
1 REM TAIL GUNNER
100 CL=30720:DEFFNR(X)=INT(RND(1)*X):DEFFNX(X)=X-7680-INT((X-7680)/22)*22:TT=30:
DL=1.8
110 POKE36879,12:PRINT"";
120 POKE36878,15
130 PRINT"TIME 00:"TT:PRINT"SCORE"SC"";
140 FORN=1TO22
150 PRINT:PRINTTAB(11)"  ";:NEXT:PRINT"                    "
160 PA=7702:TI$="000000"
170 NA=PA-22*((PEEK(197)=13)-(PEEK(197)=37))-(PEEK(197)=21)+(PEEK(197)=45)
180 IFNA<>PATHENPOKE36877,130:POKE36874,130
190 GOSUB380
200 POKE36877,0:POKE36874,0
210 IFPEEK(197)=46THENGOSUB280
220 IFFNR(INT(DL))=0THEN250
230 NA=PA+22*(FNR(3)-1)+FNR(3)-1
240 GOSUB380
250 PRINT""RIGHT$(TI$,2)
260 IFVAL(TI$)=TTTHEN420
270 GOTO170
280 NA=20:FORN=8121TO7932STEP-21:POKE36877,250:POKE36876,250
290 POKEN,78:POKEN+CL,2:POKEN+NA,77:POKEN+NA+CL,2
300 POKE36877,0:POKE36876,0:NA=NA-2:NEXT
310 NA=20:FORN=8121TO7932STEP-21
320 POKEN,32:POKEN+NA,32
330 NA=NA-2:NEXT:IFPA<>7911THENRETURN
340 T1=VAL(TI$)
350 POKE36877,130:FORN=15TO0STEP-1:POKE36878,N:FORNA=1TO50:NEXT:NEXT
360 SC=SC+TT-T1:DL=DL+.2:TT=TT-2:IFTT<10THENTT=10
370 GOTO110
380 IFNA<7702ORNA>8185ORABS(FNX(NA)-FNX(PA))>1THENNA=PA:RETURN
390 IFPEEK(PA)<>160THENPOKEPA,32
400 IFPEEK(NA)<>160THENPOKECL+NA,1:POKENA,81
410 PA=NA:RETURN
420 FORN=1TO8:POKE36876,ABS(SGN(PEEK(36876))-1)*190:FORNA=1TO50:NEXT:NEXT
430 PRINT"                    YOU RAN OUT OF TIME  ";
440 PRINT"                    "
450 PRINT"ANOTHER GAME (Y/N)"
460 POKE198,0
470 GETA$:IFA$<>"Y"ANDA$<>"N"THEN470
480 IFA$="N"THENPOKE36879,27:PRINT"";:POKE198,0:END
490 RUN
500 GOTO500
510 RETURN

READY.
```

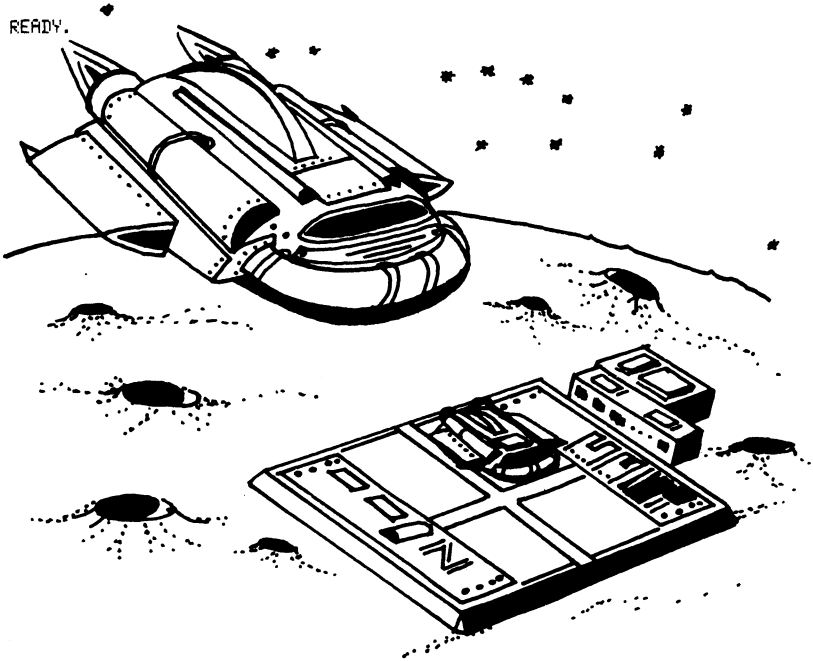
SPACE DOCK

La tua missione, dovresti deciderti ad accettarla, è di condurre il tuo "shuttle" attraverso le distese stellari e attraccarlo al molo del tuo porto sicuro. A tale scopo devi anzitutto avere il controllo dello "shuttle". Puoi mutare soltanto l'angolo di rotazione dello "shuttle". Ciò può creati delle difficoltà iniziali, ma, alla fine ci riuscirai certamente. Il punteggio che otterrai è determinato dal livello di gioco nel quale ti trovi, e dal tempo che impiegherai per attraccare al molo. Ad ogni nuova partita il livello di difficoltà aumenta.

```
1 GOTO2000
10 PRINT"[illegible]  "
20 TI$="000000"
30 POKECL+P,2:POKESC+P,RO(RO)
100 REM
110 GETA$:POKE198,0
160 N=P+R1(RO)
163 IFN=DTHENT=VAL(TI$):GOTO600
164 MM=PEEK(M)
165 IFPEEK(SC+N)=42ORN<22ORN>505GOTO500
170 POKECL+P,0:POKESC+P,32:POKECL+N,2:POKESC+N,RO(RO):P=N
172 FORX=1TO200/S/1.5:NEXT
190 IFA$="X"THENRO=RO-1
200 IFA$="Z"THENRO=RO+1
210 IFRO<1THENRO=8
220 IFRO>8THENRO=1
230 GOTO100
500 POKE36878,15:POKE36877,150:FORX=1TO40
510 D=PEEK(36879):IF(DAND8)=0THEND=D+8:GOTO530
520 D=D-8
530 POKE36879,D:FORY=1TO80:NEXT:NEXT:POKE36878,0:POKE36877,0
540 PRINT"[illegible]YOU CRASHED[illegible]"
550 PRINT:PRINT"[illegible] ANOTHER GAME(Y/N) [illegible]"
560 GETA$:IFA$="Y"THENRUN
565 IFA$<>"N"GOTO560
570 POKE36879,27:PRINT"[illegible]";:END
600 POKE36878,15:FORY=1TO4:FORX=128TO200:POKE36876,X:NEXT:NEXT:POKE36878,0:POKE3
6876,0
610 M=M+S*40-100*(S=4)-20*(T-10):IFM<0THENM=0
900 S=S+.5:IFS>4THENS=4
1000 POKE36879,14:PRINT"[illegible]";:CL=38400:SC=7680
1010 FORX=1TOS*15:R=INT(RND(1)*506):POKECL+R,1:POKESC+R,42:NEXT
1015 RESTORE
1020 FORX=1TO8:READR1(X):NEXT:FORX=1TO4:READRO(X):RO(X+4)=RO(X):NEXT
1030 DATA-22,-21,1,23,22,21,-1,-23,93,78,64,77
1035 PRINT"[illegible]PRESS ANY KEY TO PLAY"
1036 GETA$:IFA$=""GOTO1036
1040 RO=4:P=22:S=1
```

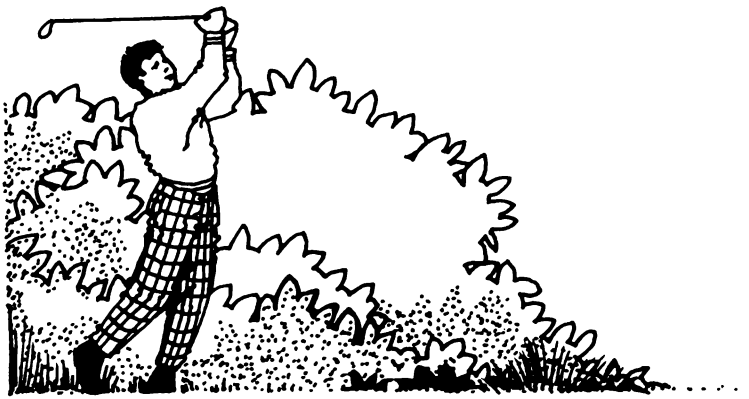
```
1050 PRINT"▒
1053 D=INT(RND(1)*43)+418
1055 POKESC+D,81:POKECL+D,2
1060 PRINT"▒▒"M
1070 GOTO10
2000 POKE36879,30:PRINT"▒▒▒▒▒▒    ▒SPACE DOCKING▒"
2010 PRINT"▒▒▒▒▒▒▒     DO YOU WANT            INSTRUCTIONS     ▒"
2020 PRINT"▒▒▒▒      [Y OR N]"
2030 GETA$:IFA$<>"Y"ANDA$<>"N"THEN2030
2035 POKE198,0
2040 IFA$="N"THEN900
2060 POKE36879,127:PRINT"▒▒   SPACE DOCKING"
2070 PRINT"▒▒  THE OBJECT OF THE   GAME IS TO GUIDE YOUR SPACE SHUTTLE TO THE"
2080 PRINT"RED DOCKING PORT.":PRINT"   YOU MUST AVOID THE WHITE STARS AND ALSO T
HE TOP AND";
2090 PRINT"BOTTOM OF THE SCREEN.IT IS SAFE TO LEAVE ONE  SIDE OF THE SCREEN AND"
;
2100 PRINT"REAPP-EAR ON THE OTHER."
2110 PRINT"▒PRESS ANY KEY"
2115 GETA$:IFA$=""GOTO2115
2117 POKE198,0
2120 PRINT"▒YOUR SCORE IS SHOWN ON THE TOP LEFT OF THE  SCREEN."
2130 PRINT"▒ CHANGE THE DIRECTION OF THE SHUTTLE WITH:-"
2140 PRINT"▒Z-ROTATE CLOCKWISE"
2145 PRINT"▒X-ROTATE ANTICLOCKWISE"
2150 PRINT"▒▒PRESS ANY KEY"
2160 GETA$:IFA$=""GOTO2160
2170 POKE198,0:GOTO900

READY.
```



GOLF

Attenzione! Ci sono 18 buche nel Campo di Golf Commodoro. Hai due gruppi di mazze a tua disposizione per spostarti nel campo sotto la pari. Il primo gruppo deve essere usato sul "tee" e in pista. Le mazze di questo gruppo sono più robuste di quelle del secondo. Le mazze dell'altro gruppo sono da usare sul prato o nelle zone adiacenti. Per scegliere la forza di ogni colpo, devi inserire un numero che vada da uno a nove.
Le mazze del primo gruppo hanno una portata massima di circa 300 metri, quelli del secondo di circa 60. Sarai in pari per ogni buca nel momento in cui ci giocherai e saprai anche il tuo totale corrente e la relativa pari. Alla fine il computer ti dirà il punteggio e la pari per le 18 buche. Quando ogni colpo è giocato, udrai la palla battere e rimbalzare a uno stop.

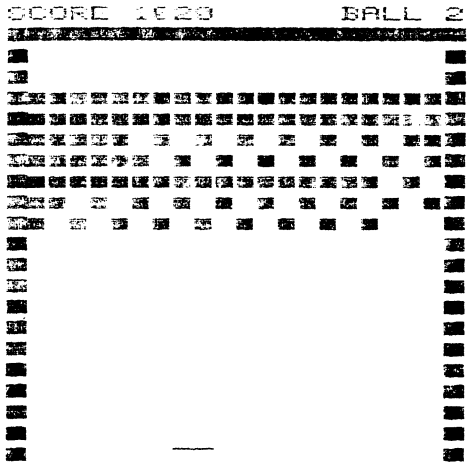


```
100 n=rnd(-ti):Poke36878,15
110 deffnr(x)=int(rnd(1)*x):Poke36879,93:Print"■■"chr$(14)
120 9osub470
130 ho=1:ts=0:tP=0
140 s=0:Print"■You are on the tee at hole"ho
150 d=fnr(250)+100:P=int(d/60):ifP<2thenP=2
160 tP=tP+P
170 Print"■■It is"d"yds lon9,":Print"with a Par of "P
180 ifs>0thenPrint"■Distance to hole:"d:Print"■Shots so far:"s:ifd>40thenPrint"■
On the fairway"
190 ifd<20thenPrint"■On the 9reen":9oto210
200 ifd<40thenPrint"■In si9ht of the 9reen"
210 Print"■Select stren9th(1-9)":Poke198,0
220 9etd$:ifd$<"1"ord$>"9"then220
230 s=s+1
240 d1=val(d$)*(25+fnr(15)):ifd<40thend1=int(d1/6)
250 ifd<5andd$<"3"thend1=d
260 ifd1=0thend1=1
261 Poke36877,150:Poke36876,200:forn=1to50:next
262 Poke36876,0:Poke36877,0:forn=1tod1*7:next
263 ifd1<20then270
264 form=1tofnr(4)+1
265 forn=1to50*(5-n):next
266 Poke36877,150:Poke36874,200:forn=1to50:next:Poke36877,0:Poke36874,0
267 next
270 Print"■You hit"d1"yds"
280 d=abs(d-d1):ifd<=1then310
290 forn=1to2000:next
300 9oto180
310 Poke36874,130
320 Print"■  ■■■  ■■■■■■■■■PLOP■■■■■■■■■
```

```
100 N=RND(-TI):POKE36878,15
110 DEFFNR(X)=INT(RND(1)*X):POKE36879,93:PRINT"■"CHR$(14)
120 GOSUB470
130 HO=1:TS=0:TP=0
140 S=0:PRINT"■ IOU ARE ON THE TEE AT HOLE"HO
150 D=FNR(250)+100:P=INT(D/60):IFP<2THENP=2
160 TP=TP+P
170 PRINT"■■ T IS"D"YDS LONG,":PRINT"WITH A PAR OF "P
180 IFS>0THENPRINT"■ ISTANCE TO HOLE:"D:PRINT"■■HOTS SO FAR:"S:IFD>40THENPRINT"■
N THE FAIRWAY"
190 IFD<20THENPRINT"■ N THE GREEN":GOTO210
200 IFD<40THENPRINT"■ N SIGHT OF THE GREEN"
210 PRINT"■■ELECT STRENGTH(1-9)":POKE198,0
220 GETD$:IFD$<"1"ORD$>"9"THEN220
230 S=S+1
240 D1=VAL(D$)*(25+FNR(15)):IFD<40THEND1=INT(D1/6)
250 IFD<5ANDD$<"3"THEND1=D
260 IFD1=0THEND1=1
261 POKE36877,150:POKE36876,200:FORN=1TO50:NEXT
262 POKE36876,0:POKE36877,0:FORN=1TOD1*7:NEXT
263 IFD1<20THEN270
264 FORM=1TOFNR(4)+1
265 FORN=1TO50*(5-N):NEXT
266 POKE36877,150:POKE36874,200:FORN=1TO50:NEXT:POKE36877,0:POKE36874,0
267 NEXT
270 PRINT"■ IOU HIT"D1"YDS"
280 D=ABS(D-D1):IFD<=1THEN310
290 FORN=1TO2000:NEXT
300 GOTO180
310 POKE36874,130
320 PRINT"■  ■■■
```

BREAKOUT

Questa è una nuova versione di un vecchio e prediletto gioco. L'obiettivo è di abbattere i mattoni di un muro per segnare punti. La tua arma è una palla che lanci con la tua mazza e che fai rimbalzare sul muro. Hai tre palle da giocare. Il gioco si conclude quando la terza palla va fuori gioco. La virgola "," sposta la tua mazza a sinistra, il segno "/" la sposta a destra.



```
100 S=0:LV=1:GOTO130
110 DY=-DY:S=S+(PEEK(30720+PP)AND15)*10:PRINT"SCORE"S
112 IFINT(S/5600)*5600=STHENGOSUB500:IFLV>1THENLV=LV-1
115 GOTO120
120 POKE36876,240:POKE36876,0:RETURN
130 D$="":R$=""
140 DEFFNR(X)=INT(RND(1)*X):POKE36878,15
150 SP$="  "
160 GOSUB410
170 GOSUB500
180 PY=11:PX=1:B=10:DX=1:DY=1
190 NB=B+(PEEK(197)=29)-(PEEK(197)=30):IFNB<1ORNB>19THENNB=B
200 PRINT"";D$;TAB(B);"  ";D$;TAB(NB);"";:B=NB
210 NY=PY+DY:IFNY=2ORNY=20THENDY=-DY:GOSUB120
220 NX=PX+DX:IFNX=1ORNX=20THENDX=-DX
230 IFNY=20ANDNX<>1ANDNX<>20THENDX=SGN(NX-B-.5)
240 IFNY=11ANDFNR(3)=1ANDNX<18ANDDY=1THENNX=PX
250 IFNY=20ANDB<>NXANDB+1<>NXTHEN300
260 PP=NY*22+NX+7680:IFPEEK(PP)=208THENGOSUB110
270 PRINT"";LEFT$(D$,PY);LEFT$(R$,PX)" "
280 PRINT"";LEFT$(D$,NY);LEFT$(R$,NX)"":PX=NX:PY=NY
290 GOTO190
300 IFLV=3THEN350
310 PRINT""D$"  LOST BALL  ";
320 LV=LV+1:PRINT""LEFT$(R$,20)LV
330 FORN=1TO1000:NEXT:PRINT""D$"  "SP$SP$SP$;
340 GOTO180
350 PRINT""LEFT$(D$,10)"GAME OVER"
360 PRINT"PRESS ANY KEY"
370 FORN=1TO1000:NEXT
380 IFPEEK(197)=64THEN380
390 RUN
400 FORN=1TO1000:NEXT:RUN
410 POKE36879,25:PRINT"  BREAKOUT  "
420 PRINT"MOVE YOUR BAT WITH-  "
430 PRINT"  , -*- / "
440 PRINT"  "
450 PRINT"LEFT SIDE OF BAT HITS  THE BALL TO THE LEFT AND THE RIGHT,TO THE  RI
GHT."
460 PRINT"PRESS ANY KEY TO PLAY"
470 FORN=1TO1000:NEXT
480 IFPEEK(197)=64THEN480
490 RETURN
500 POKE36879,8:PRINT"  ";
510 FORN=1TO20:PRINTSP$;:NEXT
520 PRINT"";
530 FORN=7TO1STEP-1:PRINTMID$("",N,1)"";:NEXT
540 PRINT"SCORE"S"BALL"LV
550 RETURN
```

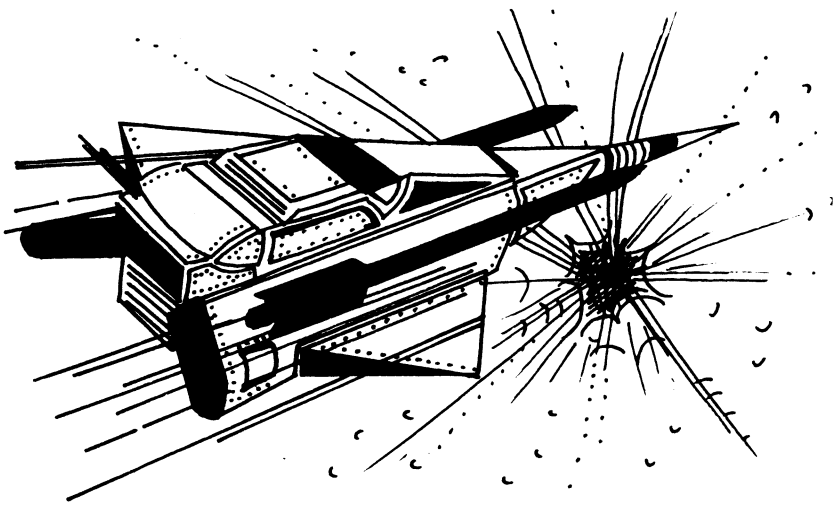
MOONBASE

Sei su una postazione-laser, sulla base lunare, e devi proteggere la città blu che è mostrata sul video da un bombardiere alieno. Sei difeso da una barriera di calcestruzzo a due livelli. L'alieno tenterà di bombardare passando attraverso la barriera per raggiungere la città in basso. La tua base serve a due scopi:
1) Per abbattere l'alieno
2) Per nascondere i buchi nella barriera di calcestruzzo
Segnare punti ti serve solo a sopravvivere, mentre quando riesci ad abbattere l'alieno riceverai un premio. Se il punteggio che segni è alto, puoi scrivere il tuo nome (fino a otto lettere). "Z" ti sposta a sinistra e "C" a destra. Premi F7 per sparare.

```
0 GOTO96
1 POKE36874,150:FORX=P-22TO7944STEP-22:CH=PEEK(X):POKEX,103:POKECL+X,7
2 IFCH=95THENSC=SC+500:POKE36874,0:GOSUB2000:D=D+.1:GOTO105
3 POKEX,32:NEXT:POKE36874,0
4 GOTO230
10 POKE36877,150:FORX=PA+22TO8164STEP22:CH=PEEK(X):POKEX,103:POKECL+X,7
15 IFCH=160THENPOKEX,32:POKE36877,0:GOTO250
20 IFCH=105ORCH=102THENGOSUB2000:GOTO3000
25 POKEX,32:NEXT
96 HI=3000:CL=38400-7680:NA$="VIC-20"
97 DEFFNX(X)=X-7680-22*INT((X-7680)/22)
98 POKE36879,9
99 LL=3:SC=0:D=.2
100 GOSUB5000
105 POKE36878,15:GOSUB1000
110 PA=7778:Q=0
120 P=8054
200 IFPEEK(197)=63THEN1
210 N=P+2*((PEEK(197)=33)-(PEEK(197)=34)):IFN<8054ORN>8074THENN=P
220 POKEP,160:POKEP+1,160:POKECL+N,2:POKEN,105:POKECL+N+1,2:POKEN+1,95:P=N
230 SC=SC+1:PRINT"■SCORE"SC
240 Q=Q+D:IFQ<1THEN200
244 POKE36876,140
245 IFRND(1)<.15THENGOTO10
246 Q=0
250 NA=PA+2*(SGN(RND(1)-RND(1)))+22*INT(RND(1)+.2)
260 IFABS(FNX(NA)-FNX(PA))>2THENNA=PA
270 IFNA>8054THENGOSUB2000:GOTO1200
280 POKEPA,32:POKEPA+1,32:POKECL+NA,4:POKENA,95:POKENA+CL+1,4:POKENA+1,105:PA=NA
290 POKE36876,0:GOTO200
997 POKE198,0
998 GETA$:IFA$=""THEN998
999 END
1000 PRINT"■■■
```

```
1010 PRINT"";
1020 PRINT"SCORE"SC:PRINT"HI"HI;NA$""LL
1030 PRINT"":RETURN
1200 PRINT"GAME OVER"
1210 IFSC>HITHENHI=SC:POKE198,0:INPUT"NAME";NA$:NA$=LEFT$(NA$,8)
1240 FORX=1TO3000:NEXT
1250 GOTO99
2000 POKE36877,180:FORX=15TO0STEP-1:POKE36878,X:FORN=1TO100:NEXT:NEXT:POKE36877,
0:RETURN
3000 LL=LL-1:IFLL>0THEN105
3010 GOTO1200
5000 PRINT"        MOONBASE"
5010 PRINT"   YOUR MISSION IS TO PROTECT YOUR MOONBASE ().FROM THE ALIEN"
5020 PRINT"BOMBER HOVERING ABOVE"
5030 PRINT"THE MOONBASE IS PROTE-CTED BY TWO SHEILDS   ().YOU CAN HELP IN
 TWO WAYS-"
5040 PRINT"  1) REPAIR THE TOP   SHIELD BY MOVING YOUR CANNON()OVER IT"
5050 PRINT"  2) BY DESTROYING THE ALIEN WHEN IT IS LOW ENOUGH"
5060 PRINT"    PRESS ANY KEY":POKE198,0
5070 GETA$:IFA$=""THEN5070
5080 PRINT"TO MOVE -"
5090 PRINT"LEFT-":PRINT"RIGHT-":PRINT"FIRE-F7"
5100 PRINT"THE LONGER YOU SURVIVE,THE MORE YOU SCORE.  500 POINTS ARE SCORED FO
R ";
5110 PRINT"HITTING THE ALIEN"
5120 PRINT" A LIFE IS LOST IF THE ALIEN HITS THE MOON- BASE()OR YOUR CANNON"
5130 PRINT" THE GAME ENDS IF THE     ALIEN LANDS"
5140 PRINT"PRESS ANY KEY TO PLAY"
5150 POKE198,0
5160 GETA$:IFA$=""THEN5160
5170 RETURN

READY.
```



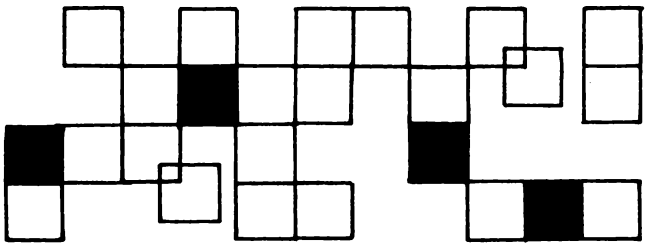
LIFE

Questa è una delle più famose simulazioni eseguite da un computer, in cui l'evoluzione di una colonia di cellule è creata sul video. La vita, la morte e la nascita delle cellule seguono le regole formulate da John Conway, l'inventore del gioco.

1) Se una cellula è circondata da meno di due altre cellule, morirà di solitudine
2) Se una cellula ha due altre cellule vicine, morirà alla prossima generazione
3) Se uno spazio vuoto è circondato da tre cellule, una nuova cellula nascerà in quello spazio nella prossima generazione.
4) Se una cellula ha quattro o più cellule vicine, morirà di sovraffollamento.

Dà il tuo modello di cellule all'inizio del gioco. Se vuoi aggiungere una cellula, scrivi la colonna e la fila nelle quali desideri collocarla. Se cambi idea, puoi cancellare la cellula scrivendo di nuovo le coordinate. Scrivi "Ø" per il numero "orizzontale" quando vuoi incominciare il processo della VITA. Hai tre controlli che puoi usare durante l'evoluzione delle generazioni (i tasti devono essere tenuti abbassati finché la reazione è visualizzata):

I controlli e i loro effetti sono i seguenti:
FI — Ritardi lo sviluppo generazionale finché il tasto non è rilasciato. È utile se desideri studiare un certo modello di cellule.
F3 — Cancelli tutte le cellule per ricominciare il gioco.
F5 — Ricominci il gioco lasciando intatto l'attuale modello di cellule.



LIFE

```
1 REM LIFE
100 POKE36879,93:PRINT"■";
110 DIMM(1,10,10)
120 G=0
130 GOSUB250
140 INPUT"■ACROSS";X:IFX=0THEN180
150 IFX<1ORX>9THEN140
160 INPUT"■DOWN";Y:IFY<1ORY>9THEN160
170 M(1,X,Y)=ABS(M(1,X,Y)-1):GOTO130
180 FORY=1TO9:FORX=1TO9:C=M(0,X-1,Y-1)+M(0,X-1,Y)+M(0,X-1,Y+1)+M(0,X,Y-1)+M(0,X,
Y+1)
190 C=C+M(0,X+1,Y-1)+M(0,X+1,Y)+M(0,X+1,Y+1)
200 IFC=3THENM(1,X,Y)=1
210 IFC<2ORC>3THENM(1,X,Y)=0
220 NEXT:NEXT
230 G=G+1:GOSUB250:GOTO180
240 END
250 IFG<2THENPRINT"■";
260 PRINT"■■■   LIFE   ■"
270 IFG>0THENPRINT"■■■ GENERATION: ■"G
280 PRINT"■■■■ 123456789 ■"
290 FORY=1TO9:X$="■■"+RIGHT$(STR$(Y),1)+"■":PRINTX$;
300 FORX=1TO9:IFM(1,X,Y)=1THENPRINT"■■";:GOTO320
310 PRINT"■■";
320 M(0,X,Y)=M(1,X,Y):NEXT:PRINTX$:NEXT
330 PRINT"■■■ 123456789 ■"
340 C=PEEK(197):IFP<>64THEN360
350 RETURN
360 POKE198,0:IFC=39THEN340
370 IFC=47THENRUN
380 IFC=55THEN120
390 GOTO350

READY.
```

FRUIT MACHINE

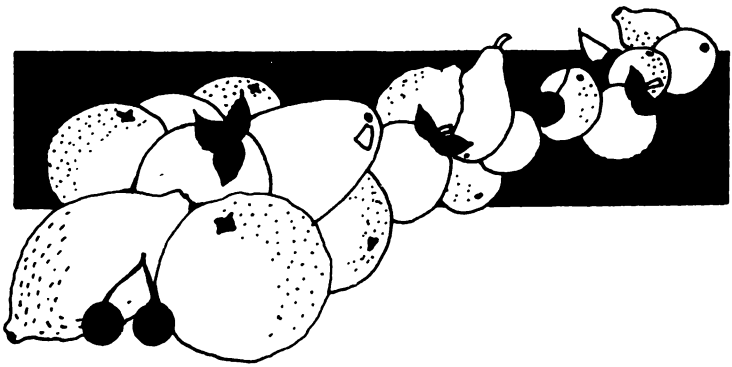
Il programma simula la "fruit machine", inizi con dieci monete. Queste sono esposte nel "vaso" in fondo al video. Ogni colpo ti costa una moneta, e puoi vincere in due modi:

1) Due cose uguali, come CAMPANA - CAMPANA- QUALSIASI COSA
2) Tre cose uguali, come SPADA-SPADA-SPADA

L'ammontare della vincita dipende dal frutto vincente. La STRISCIA rende di più, la CILIEGIA è quella che rende meno. Il denaro della vincita è versato nel vaso in fondo al video. Il gioco si conclude se il vassoio si riempie o si svuota. Ci sono due cose che possono aiutarti:

1) HOLD — Quando ciò accade puoi fermare qualsiasi o tutte le "ruote" premendo i numeri da uno a tre (la "ruota" di sinistra è 1). Puoi cancellare un "hold" su una "ruota" premendo il numero della "ruota" una seconda volta. Batti la barra spaziatrice quando vuoi ricominciare il gioco.
2) NUDGE — Quando ciò accade, ti saranno offerte da una a tre mosse. Per muovere una "ruota", usa lo stesso tasto che hai usato per HOLD. Se non desideri usare tutte le tue mosse, premi SPAZIO.

Nota che la BARRA SPAZIATRICE è usata come pulsante di "start".



FRUIT MACHINE

```
1 REM FRUIT MACHINE
100 DEFFNR(X)=INT(X*RND(1))
110 POKE36879,56:POKE36878,15
120 PRINT"■■     FRUIT MACHINE    ■";
130 PRINT"■■■
```

```
690 PRINT"[illegible] YOU [illegible] BROKE [illegible] THE [illegible]BANK [illegible]"
700 GOTO730
710 PRINT"[illegible]";:POKE36879,42
720 PRINT"[illegible] GAME OVER [illegible]"
730 FORN=1TO1000:NEXT
740 PRINT"[illegible]PRESS ANY KEY TO PLAY"
750 IFPEEK(197)=64THEN750
760 POKE198,0:RUN
770 DATA"[illegible]"
780 DATA"[illegible]"
790 DATA"[illegible]"
800 DATA"[illegible]"
810 DATA"[illegible]"
820 DATA"[illegible]"
830 DATA"[illegible]"
840 DATA"[illegible]BAR [illegible]"
850 PRINT"[illegible]";:FORN=1TO3:A$="    ":IFH(N)=1THENA$="[illegible]HOLD[illegible]"
860 PRINTA$"[illegible]";:NEXT:RETURN
870 FORN=1TO8:POKE36876,ABS(SGN(PEEK(36876))-1)*240:FORN1=1TO20:NEXT:NEXT:RETURN

READY.
```

SPOTS FROM OUTER SPACE

In questo gioco stile Invasori (sembra che si tratti di una malattia particolarmente contagiosa) devi proteggere il tuo pianeta natale da macchie invasori. Se sei colpito da un missile lanciato dalle macchie, o se la macchie atterrano, sarai ucciso. In questo caso o se vinci tutte le macchie, il gioco ricomincerà automaticamente. Il tuo laser-base è controllato dai tasti "Z" = sinistra, "X" = destra, e "*" = per sparare.

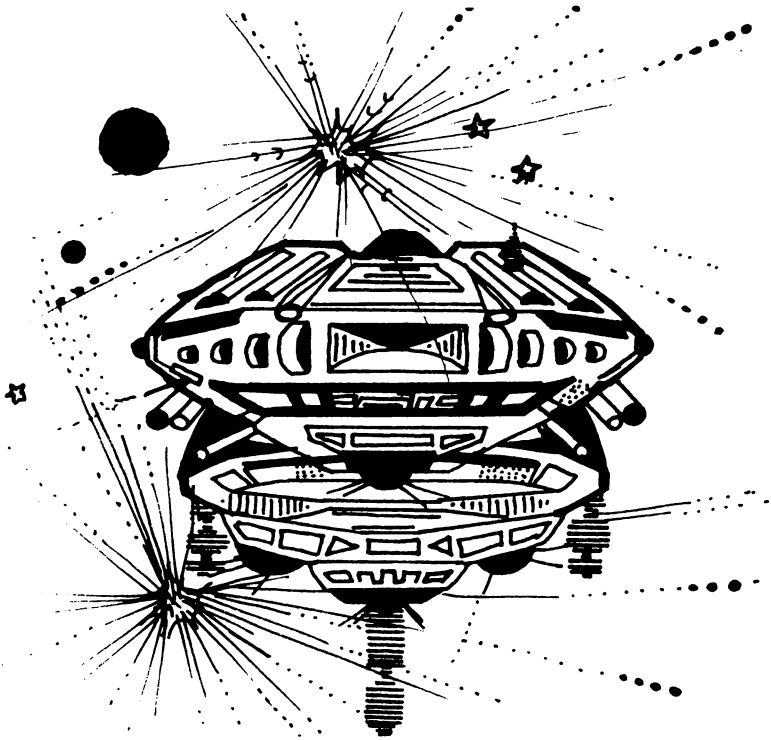
```
1 REM SPOTS FROM OUTER SPACE
5 POKE650,255
10 PRINT"■"
15 F=38
16 M=12
20 POKE36867,34
25 H=INT(RND(1)*22)+7790
30 POKE36879,42
31 POKE198,0
32 PRINT"■■SPOTS FROM OUTER SPACE"
33 PRINT"■■SKILL LEVEL?":PRINT"1-40"
34 INPUTG
38 PRINT"■"
40 A=7998
50 PRINT"■● ● ● ● ● ● ● ● ● ● ●"
60 PRINT"■● ● ● ● ● ● ● ● ● ● "
70 PRINT"■● ● ● ● ● ● ● ● ● ● ●"
80 PRINT"■● ● ● ● ● ● ● ● ● ● "
200 REM GENERAL ROUTINE
210 POKEA,65
220 D=INT(RND(1)*G)+1
230 IFD=1 THENF=F+1:POKE36865,F
231 IFD=1THENA=A-22:POKEA+22,96
232 IFD=1THENM=M+INT(RND(1)*3)-1
233 POKE36864,M
234 POKEH,96:H=H+22:POKEH,93
236 IFH>8010THENH=INT(RND(1)*22)+7790
237 IFPEEK(H+22)=65THEN700
240 IFPEEK(A)=81THEN700
245 IFA<7680THEN700
250 REM CONTROL
260 GETA$
270 IFA$="Z"THENA=A-1:POKEA+1,32
280 IFA$="X"THENA=A+1:POKEA-1,32
290 IFA$="*"THEN500
300 GOTO200
500 REM FIRE
510 B=A-22
```

```
520 POKEB,32:B=B-22
530 POKEB,34
535 POKEH,96:H=H+22:POKEH,93
536 IFH>8032THENH=INT(RND(1)*22)+7790
537 IFPEEK(H+22)=65THEN700
540 IFPEEK(B-22)=81THEN580
542 IFB=HTHENH=INT(RND(1)*22)+7790
543 IFB=HTHEN200
545 IFB<7678THEN200
550 GOTO520
580 POKEB,32:POKEB-22,32
582 C=C+1:IFC=42THEN600
590 GOTO200
600 PRINT"■WELL DONE!!"
610 POKE36867,46
620 POKE36865,38
630 GOTO720
700 PRINT"THEY'VE KILLED YOU!!!!"
710 POKE36867,46:POKE36865,38
720 FORM=1TO2000:NEXT:RUN

READY.
```

DESTROYER

Questo programma è stato scritto da Adam Burbidge. Una astronave si sposterà da sinistra verso destra passando sopra lo spazio profondo. La tua missione è di distruggere gli alieni ostili che sono in grado di far saltare in aria la tua astronave fino a disintegrarla completamente, se non userai tutta la tua attenzione. Un alieno può essere distrutto solo nel caso che si trovi fra cinque spazi direttamente di fronte a te. I punti sono segnati a seconda del colore degli alieni. "A" sposta la tua astronave verso il basso, "B" verso l'alto; quando vuoi sparare, premi "P".
Attenzione: gli alieni si spostano molto velocemente e la loro avanzata tecnologia permette loro di diventare invisibili di quando in quando. Anche da invisibili possono essere pericolosi.



```
100 DIMZ(5),A(9)
110 REM**DESTROYER
120 REM**ADAM BURBIDGE
130 I=3:D=INT(RND(1)*10)
140 K(1)=0:G=7900
150 PRINT"□"
160 B=B+1
170 C=INT(RND(1)*506)+7680
180 A(B)=C
190 D=D-1:IFD<1THEN210
200 GOTO160
210 POKE36879,8
220 POKEG,62
230 E=0
240 F=A(E)
250 E=E+1
260 POKEF,35
270 Z(2)=INT(RND(1)*8)
280 POKEF+30720,Z(2)
290 IFE=9THEN310
300 GOTO240
310 Z(0)=INT(RND(1)*7)
320 GOSUB1070
330 POKEG,32:FORT=1TO100:NEXT
340 POKEG,32:FORT=1TO100:NEXT
350 G=G+1:K(0)=K(0)+1
360 IFK(0)>22THENK(0)=0
370 IFPEEK(G-22)=35THEN920
380 IFPEEK(G+1)=35THEN920
390 GETH$
400 IFG<7680THENG=8165
410 IFG>8165THENG=7680
420 IFH$=""THEN470
430 IFH$="A"THENG=G+22
440 IFH$="D"THENG=G-22
450 POKEG,62
460 IFH$="P"THEN620
470 POKEG,62
480 GOTO490
490 B=INT(RND(1)*9)
500 POKEA(B),32
510 C=INT(RND(1)*4)+1
520 IFC=4THENA(B)=A(B)+22
530 IFC=3THENA(B)=A(B)-22
540 IFC=2THENA(B)=A(B)-1
550 IFC=2THENA(B)=A(B)+1
560 IFA(B)<7680THENA(B)=7680
570 IFA(B)<GTHENA(B)=A(B)+22
580 IFA(B)>GTHENA(B)=A(B)-22
590 POKEA(B)+30720,INT(RND(1)*7)
600 POKEA(B),35
610 GOTO330
620 IFPEEK(G+1)=35THENPOKEG+1,102:GOTO760
630 IFPEEK(G+1)=218THENPOKEG+1,102:SC=SC+20:GOTO700
640 IFPEEK(G+2)=35THENPOKEG+2,102:GOTO780
650 IFPEEK(G+2)=218THENPOKEG+2,102:SC=SC+20:GOTO720
660 IFPEEK(G+3)=35THENPOKEG+3,102:GOTO800
670 IFPEEK(G+4)=35THENPOKEG+4,102:GOTO820
680 IFPEEK(G+5)=35THENPOKEG+5,102:GOTO840
690 GOTO330
```

```
700 SC=SC+PEEK(G+30721)*100
710 GOTO850
720 SC=SC+PEEK(G+30722)*100
730 GOTO850
740 SC=SC+PEEK(G+30723)*100
750 GOTO850
760 SC=SC+PEEK(G+30721)*10
770 GOTO850
780 SC=SC+PEEK(G+30722)*10
790 GOTO850
800 SC=SC+PEEK(G+30723)*10
810 GOTO850
820 SC=SC+PEEK(G+30724)*10
830 GOTO850
840 SC=SC+PEEK(G+30725)*10
850 POKE36877,128
860 POKE36878,15
870 A(B)=0
880 FORT=1TO300:NEXT
890 POKE36877,0
900 SC=SC+10
910 GOTO330
920 POKEG,102
930 POKE36877,220
940 FORL=15TO0STEP-1
950 POKE36876,INT(RND(1)*128)+128
960 POKE36878,L
970 FORM=1TO100
980 NEXT
990 NEXT
1000 POKE36877,0
1010 POKE36876,0
1020 PRINT"■■SCORE:"SC
1030 I=I-1
1040 IFI=0THENPRINT"XXX■  GAME OVER   ■":CLR:I=3:D=INT(RND(1)*10):FORT=1TO1000:N
EXT
1050 FORT=1TO1000:NEXT
1060 GOTO140
1070 Z(1)=INT(RND(1)*22)+8143
1080 Z(3)=INT(RND(1)*3)+4
1090 POKEZ(1),218
1100 POKEZ(1)+30720,Z(3)
1110 RETURN

READY.
```

CAR RACE

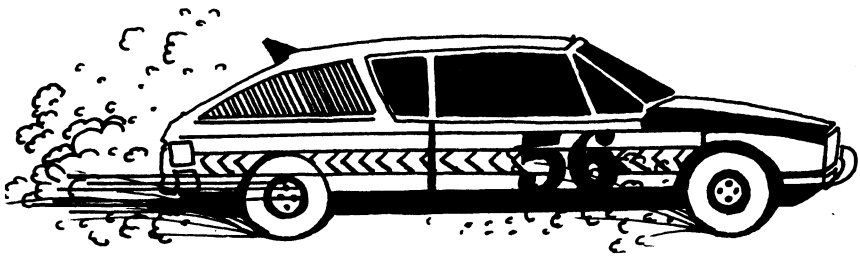
Il gioco di Kevin Smith ti mette al volante, devi guidare la tua macchina fin quando sei in grado di farlo, e cioè fino al momento in cui la tua attenzione vacilla e lo scontro è inevitabile.
I controlli sono:
"Z" = destra
"X" = sinistra

Cursore verso il basso = accelerazione
Cursore verso destra = freno
Prima marcia = F1
Seconda = F3
Terza = F5
Quarta = F7

```
1 PRINT" CAR RACE",,"Z=LEFT:X=RIGHT"
2 PRINT"GEARS",,"F1=FIRST:F7=FOURTH":PRINT"ROAD WIDTH?"
3 PRINT"[1,2,3,4]
4 GETC$:IFC$=""THEN4
5 C=7690:B=8
6 POKE36879,40
7 I=100
8 POKE650,255
10 PRINT"J"
19 K=36874
20 A=INT(RND(1)*3)-1
21 POKEK,94
22 POKEK,0
23 POKEK,94
24 POKEK,0
25 POKEK,INT(RND(1)*10)+186
50 E=C+D
51 POKEK,0
52 POKE36878,15
53 POKEK,189+INT(RND(1)*6)
60 POKEE,88
62 POKE36874,0
65 GETB$
70 IFB$="Z" THEND=D-1
75 IFB$="X" THEND=D+1
78 IFB$="C" THEN81
79 GOTO82
81 GETB$:IFB$=""THEN81
82 IFB$="F"THENI=100
84 IFB$="G"THENI=70
86 IFB$="H"THENI=55
88 IFB$="I"THENI=25
90 IFB$="M"THEN500
92 IFB$="N"THENI=I-5
100 B=B+A
```

```
102 POKEK,93
110 IFB<1THENB=B+1
111 IFB>15THENB=B-1
115 G=G+1
116 POKEK,0
117 IFC$="1"THENPRINTTAB(B)"▒▒▒▒ ▒▒▒"
118 IFC$="2"THENPRINTTAB(B)"▒▒▒▒  ▒▒▒"
119 IFC$="3"THENPRINTTAB(B)"▒▒▒▒   ▒▒▒"
120 IFC$="4"THENPRINTTAB(B)"▒▒▒▒    ▒▒▒"
121 POKE36878,15:POKE36874,193
123 IFPEEK(E)=102THEN300
125 POKEE,88
126 POKE36874,0
128 POKEK,192
129 POKEK,0
130 FORT=1TOI:NEXT
132 POKEK,95
150 GOTO20
300 POKE36878,12
305 POKE36874,0
310 FORL=1TO10
320 FORM=250TO240STEP-1
330 POKE36876,M:NEXTM
340 FORM=240TO250
350 POKE36876,M:NEXTM
360 POKE36876,0:NEXTL
370 POKE36878,0
375 PRINT"Ⓙ"
380 PRINT"YOU KEPT GOING FOR ";G/40;" MILES!!"
390 FORM=1TO2000:NEXT:RUN
500 I=I+35
505 POKE36878,13
510 FORL=235TO245
520 POKE36876,L
530 FORM=1TO10:NEXTM
540 NEXTL
550 POKE36876,0
560 GOTO100

READY.
```



SUB CHASE

Sei il comandante del HMS Vittoria, un cacciatorpediniere della classe CBM e la tua missione è di distruggere i sommergibili nemici che sono in agguato sotto di te. Sei armato con una scorta illimitata di bombe di profondità; ma stai all'erta, i sottomarini possono difendersi con i loro missili. Il totale dipende dal numero di sottomarini affondati e dal numero di colpi messi a segno dai sottomarini. I due totali sono messi in evidenza sulla parte superiore del video. Usa i seguenti caratteri di controllo per la tua nave:

"Z" — Per spostarti a sinistra
"C" — Per spostarti a destra
"B" — Per sganciare le bombe di profondità
Nota che può apparire una sola bomba alla volta.

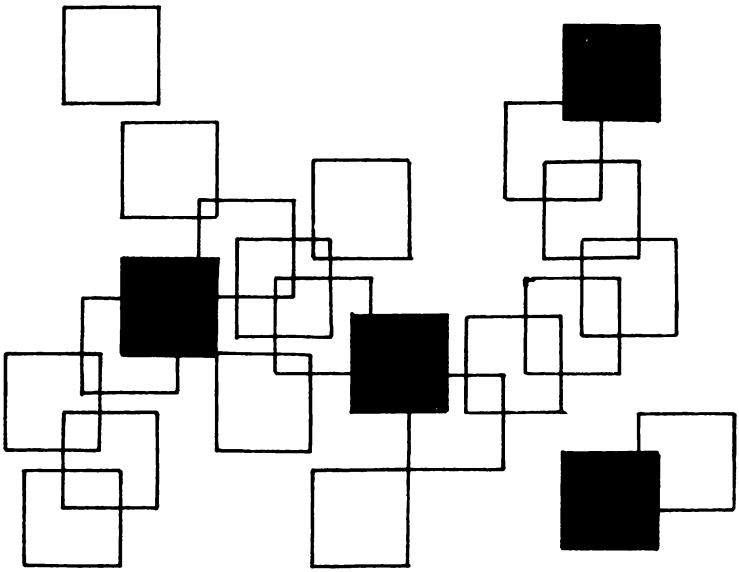
```
1 REM*SUB CHASE
2 REM*K.STAPLES
20 S3=36876:S2=36875:S=36877:V=36878:OS$="                    "
100 PRINT"□CHOOSE HOW HARD YOU   ■WANT THE GAME(8-1)    ■1 IS THE HARDEST"
110 INPUT"■CHOOSE GAME";G:IFG<1ORG>8THEN110
120 PRINT"□■  *** SUB CHASE ***"
130 PRINT"Z-LEFT.C-RIGHT.B-D/CS■"
140 FORI=1TO374:POKE7811+I,102:POKE38531+I,3:NEXT
150 GOSUB1000:GOSUB2000:U(1)=T3:DI(1)=T2:GOSUB2000:U(2)=T3:DI(2)=T2
160 IFPEEK(197)=34THENA=A+1:GOSUB1000
170 IFPEEK(197)=33THENA=A-1:GOSUB1000
180 IFDI(1)=1THEN230
190 U(1)=U(1)+1:T(1)=T(1)+1
200 IFT(1)=20THENU=U(1):GOSUB2100:T(1)=0:GOSUB2000:U(1)=T3:DI(1)=T2
210 POKEU(1)-2,102:POKEU(1)-1,60:POKEU(1),149:POKEU(1)+1,62
220 POKEU(1)-23,102:POKEU(1)-22,93:GOTO500
230 U(1)=U(1)-1:T(1)=T(1)+1
240 IFT(1)=20THENU=U(1):GOSUB2100:T(1)=0:GOSUB2000:U(1)=T3:DI(1)=T2
250 POKEU(1)-1,60:POKEU(1),149:POKEU(1)+1,62:POKEU(1)+2,102
260 POKEU(1)-22,93:POKEU(1)-21,102
500 IFDC=1THENC1=C1+22:GOTO520
510 IFPEEK(197)=35THENDC=1:C1=7836+A
520 IFDC=1THENPOKEC1,81:POKEC1-22,102:IFC1>8164THENPOKEC1,102:DC=0
530 IFU(1)=C1THENU=U(1):GOSUB3000:GOSUB3600
540 IFU(2)=C1THENU=U(2):GOSUB3000:GOSUB3600
600 IFMR=1THENM=M-22:GOTO620
610 GOSUB4000
620 IFMR=1THENPOKEM,42:POKEM+22,102:IFM<7811THENPOKEM,32:MR=0
630 IFM=7792+AORM=7791+AORM=7793+ATHENU=M:GOSUB3000:U=M+88:GOSUB5000
780 IFDI(2)=1THEN830
790 U(2)=U(2)+1:T(2)=T(2)+1
800 IFT(2)=20THENU=U(2):GOSUB2100:T(2)=0:GOSUB2000:U(2)=T3:DI(2)=T2
810 POKEU(2)-2,102:POKEU(2)-1,60:POKEU(2),149:POKEU(2)+1,62
820 POKEU(2)-23,102:POKEU(2)-22,93:GOTO999
```

```
830 U(2)=U(2)-1:T(2)=T(2)+1
840 IFT(2)=20THENU=U(2):GOSUB2100:T(2)=0:GOSUB2100:U(2)=T3:DI(2)=T2
850 POKEU(2)-1,60:POKEU(2),149:POKEU(2)+1,62:POKEU(2)+2,102
860 POKEU(2)-22,93:POKEU(2)-21,102
870 IFMS=1THENM1=M1-22:GOTO890
880 GOSUB4100
890 IFMS=1THENPOKEM1,42:POKEM1+22,102:IFM1<7811THENPOKEM1,32:MS=0
900 IFM1=7792+AORM1=7791+AORM1=7793+ATHENU=M1+88:GOSUB7000
999 GOTO160
1000 POKEV,15:FORJ=221TO244:POKES2,J
1003 FORJ1=1TO5:NEXTJ1,J:POKES2,0:POKEV,0
1005 IFA<0THENA=0
1010 IFA>17THENA=17
1020 PRINT"▒▒▒▒"TAB(A)"   | "
1030 PRINT"▒▒▒▒▒"TAB(A)"  ▒ ▒ "
1040 PRINT"▒▒▒▒▒▒"TAB(A)" ▒▒ ▒▒ "
1050 RETURN
2000 T2=INT(RND(1)*2)+1:IFT2=1THEN2030
2020 T3=7857+(INT(RND(1)*13)+1)*22:RETURN
2030 T3=7876+(INT(RND(1)*13)+1)*22:RETURN
2100 POKEU-2,102:POKEU-1,102:POKEU,102:POKEU+1,102:POKEU+2,102
2120 POKEU-23,102:POKEU-22,102:POKEU-21,102:POKEU-44,102:RETURN
3000 POKEU-1,77:POKEU,93:POKEU+1,78:POKEU-22,93
3510 POKE36879,47:POKES,220
3520 FORJ=15TO0STEP-.05:POKEV,J:NEXT
3530 POKES,0:POKE36879,27:RETURN
3600 POKEV,15:POKES,254
3610 POKEU-1,60:POKEU,149:POKEU+1,62
3620 POKEU-23,102:POKEU-22,93:POKEU-21,102:POKEU-44,102
3630 IFU>8164THENGOSUB3000:GOSUB2100:GOSUB2000:GOTO3650
3640 U=U+22:GOTO3600
3650 IFU(1)=C1THENU(1)=T3:DI(1)=T2:DC=0:C1=0:T(1)=0:SS=SS+1:PRINT"▒▒▒SUBS"SS:RET
URN
3660 IFU(2)=C1THENU(2)=T3:DI(2)=T2:DC=0:C1=0:T(2)=0:SS=SS+1:PRINT"▒▒▒SUBS"SS
4000 R1=INT(RND(1)*9)+1:IFR1>GTHENMR=1:M=U(1)
4020 RETURN
4100 R1=INT(RND(1)*9)+1:IFR1>GTHENMS=1:M1=U(2)
4120 RETURN
5000 POKEV,15:POKES,254
5010 PRINT"▒▒▒▒"OS$:PRINT"▒▒▒▒▒"OS$:PRINT"▒▒▒▒▒▒"OS$
5020 POKEU-1,95:POKEU,160:POKEU+1,105:POKEU-22,160:POKEU-44,93
5030 POKEU-23,102:POKEU-21,102:POKEU-66,102
5040 IFU>8164THENGOSUB3000:GOTO5060
5050 U=U+22:GOTO5020
5060 GOSUB2100
5070 A=0:DS=DS+1:PRINT"▒▒▒▒▒▒▒▒▒▒▒SHIPS"DS:GOSUB1000:MR=0:M=0:RETURN
6000 POKEV,15:POKES3,220:FORJ=1TO5:NEXT:POKES3,0:POKEV,0:RETURN
7000 POKEV,15:POKES,254
7010 PRINT"▒▒▒▒"OS$:PRINT"S▒▒▒▒"OS$:PRINT"▒▒▒▒▒▒"OS$
7020 POKEU-1,95:POKEU,160:POKEU+1,105:POKEU-22,160:POKEU-44,93
7030 POKEU-23,102:POKEU-21,102:POKEU-66,102
7040 IFU>8164THENGOSUB3000:GOTO7060
7050 U=U+22:GOTO7020
7060 GOSUB2100
7070 A=0:DS=DS+1:PRINT"▒▒▒▒▒▒▒▒▒▒▒SHIPS"DS:GOSUB1000:MS=0:M1=0:RETURN

READY.
```

REVERSI

In questo programma (spesso chiamato Othello), scritto da Mike O'Neill, giochi contro il computer. La battaglia che ti si presenta davanti non è delle più facili. Il programma fa uso di semplici ma efficaci algoritmi dove ogni quadrato è collocato secondo un ordine di precedenza che va da 1 a 60. Il computer osserva ogni quadrato sul quadro, in ordine di precedenza, finché ne trova uno nel quale è possibile muovere. Il programma controlla che tu non imbrogli e sa quando è possibile muovere.
Se il computer ha già mosso, ma tu non puoi farlo, muoverà due volte. Se tu hai appena mosso, e non ci sono mosse per il computer, avrai diritto a un'ulteriore mossa. Questo succede solo raramente. Se nessuno dei due giocatori può muovere, il gioco finirà. A conclusione del gioco, il punteggio finale viene indicato e puoi decidere di incominciare una nuova partita o di terminare il programma.



```
1 S2=36875:VV=36878:POKE36879,88:PRINT"J"
10 DIMB%(8,8),AX(8),AY(8),AX%(60),AY%(60)
11 FORI=1TO60:READAX%(I),AY%(I):NEXT
12 DATA1,1,1,8,8,8,8,1,3,1,6,1,8,3,8,6,6,8,3,8,1,6,1,3,3,3,6,3,6,6,3,6
13 DATA4,1,5,1,8,4,8,5,5,8,4,8,1,5,1,4,4,3,5,3,6,4,6,5,5,6,4,6,3,5,3,4
14 DATA4,2,5,2,7,4,7,5,5,7,4,7,2,5,2,4,3,2,6,2,7,3,7,6,6,7,3,7,2,6,2,3
15 DATA7,1,8,2,8,7,7,8,2,8,1,7,1,2,2,1,7,2,7,7,2,7,2,2
17 FORI=1TO8:READAX(I),AY(I):NEXT
18 DATA1,-1,1,0,1,1,0,1,-1,1,-1,0,-1,-1,0,-1
20 FORI=1TO8:FORJ=1TO8:B%(I,J)=0:NEXTJ,I
21 B%(4,4)=1:B%(5,5)=1:B%(4,5)=2:B%(5,4)=2:SS=0
25 GOSUB8000:GOSUB1000
98 PRINT"DO YOU WANT TO":INPUT"GO FIRST(Y OR N)";A$:IFA$="N"THEN200
99 GOSUB9999:GOTO110
100 GOSUB5000
110 GOSUB2000
111 P=7770+X*2+Y*44:IFB%(X,Y)<>0THENGOSUB6500:GOTO110
112 POKEP+30720,0:POKEP,81:SX=X:SY=Y:IS=1:TK=1:GOSUB3000
130 IFV=0THENPOKEP,32:GOSUB6500:GOTO110
140 B%(SX,SY)=2:GOSUB9500
200 GOSUB9999:DD%=0:FORJ=1TO60
220 SX=AX%(J):SY=AY%(J):I1=0:TK=0:IS=2:GOSUB3000:IFV=1THEN300
250 NEXT:GOSUB9000:GOTO100
300 P=7770+SX*2+SY*44:POKE30720+P,1
305 POKES2,225:FORI1=1TO10
310 POKEVV,15:POKEP,32:FORI2=1TO50:NEXT
315 POKEVV,0:POKEP,81:FORI2=1TO50:NEXT
320 NEXT
340 TK=1:IS=2:GOSUB3000:B%(SX,SY)=1:GOSUB9500:GOTO100
1000 POKE38674,1:POKE7954,81
1010 POKE38676,0:POKE7956,81
1020 POKE38718,0:POKE7998,81
1030 POKE38720,1:POKE8000,81
1200 RETURN
2000 GOSUB9999
2005 DD%=0
2010 INPUT"COLUMN";X
2020 INPUT"   ROW";Y
2030 IFX<1ORX>8ORY<1ORY>8THENGOSUB6500:GOTO2000
3000 V=1:IFB%(SX,SY)<>0THENV=0:RETURN
3010 T0=0
3020 FORI=1TO8
3030 T1=-1:TX=SX:TY=SY
3040 TX=TX+AX(I):TY=TY+AY(I):T1=T1+1:IFTX<1ORTX>8ORTY<1ORTY>8THEN3200
3050 IFB%(TX,TY)=0THEN3200
3060 IFB%(TX,TY)=ISTHEN3040
3070 T0=T0+T1
3080 IFTK=0ORT1=0THEN3200
3090 TX=TX-AX(I):TY=TY-AY(I)
3100 IFTX=SXANDTY=SYTHEN3200
3110 B%(TX,TY)=3-B%(TX,TY)
3120 I1=7770+TX*2+TY*44
3130 IFIS=1THENPOKE30720+I1,0:GOSUB7000
3140 IFIS=2THENPOKE30720+I1,1:GOSUB7500
3190 GOTO3090
3200 NEXT
3210 IFT0=0THENV=0
3220 RETURN
5000 FORSX=1TO8:FORSY=1TO8
5010 TK=0:IS=1:GOSUB3000
```

```
5020 IFV<>0THENRETURN
5030 NEXTSY,SX
5040 IFDD%=1THEN10000
5045 DD%=1:PRINT"■    YOU CAN'T MOVE":PRINT"     I'LL GO AGAIN    "
5050 FORI=1TO3000:NEXT:GOTO200
6500 POKES2,195:FORI1=1TO10
6510 PRINT"ILLEGAL MOVE"
6520 POKEVV,15:FORI2=1TO50:NEXT:PRINT"□                □":POKEVV,0:FORI2=1TO50:NEXTI
2,I1
6550 RETURN
7000 POKES2,0:POKEVV,15:FORI2=195TO240STEP0.4:POKES2,I2:NEXT:POKEVV,0:RETURN
7500 POKES2,0:POKEVV,15:FORI2=240TO195STEP-0.4:POKES2,I2:NEXT:POKEVV,0:RETURN
8000 PRINT"□■■■■■    1 2 3 4 5 6 7 8"
8010 PRINT"    ┌─┬─┬─┬─┬─┬─┬─┬─┐"
8020 A$="│ │ │ │ │ │ │ │ │"
8030 B$="    ├─┼─┼─┼─┼─┼─┼─┼─┤"
8040 FORI=1TO7:PRINTI;A$:PRINTB$:NEXT
8050 PRINT" 8 ";A$:PRINT"    └─┴─┴─┴─┴─┴─┴─┴─┘"
8200 RETURN
9000 IFDD%=1THEN10000
9010 PRINT"■    I CAN'T MOVE":PRINT" YOU HAVE ANOTHER GO":FORI=1TO1000:NEXT:DD%=
1:RETURN
9500 SS=SS+1:IFSS<60THENRETURN
9650 I1=0:I2=0
9660 FORI=1TO8:FORJ=1TO8
9670 IFB%(I,J)=1THENI1=I1+1
9680 IFB%(I,J)=2THENI2=I2+1
9690 NEXTJ,I
9695 GOSUB9999
9700 IFI1>I2THEN9750
9710 IFI2>I1THEN9800
9730 PRINT"■    IT'S A DRAW !!":GOTO9900
9750 PRINT"■   I WIN  ";I1;"TO";I2:GOTO9900
9800 PRINT"■  YOU WIN  ";I2;"TO";I1:GOTO9900
9900 INPUT"■■■ANOTHER GAME(Y/N)   ■■■";A$:IFASC(A$)=78THENPRINT"□":POKE36879,27:
END
9905 IFASC(A$)=89THENRESTORE:CLR:GOTO1
9910 GOTO9900
9995 PRINT"□":END
9999 FORI=7680TO7745:POKEI,32:NEXT:RETURN
10000 PRINT"■NEITHER OF US CAN MOVE      END OF GAME    ":FORI=1TO3000:NEXT:GOTO
9650

READY.
```

APOLLO II

In questo gioco di grafica di grande effetto, sei il comandante del modulo di atterraggio VIC-20-CBM e lo devi guidare verso il posto di atterraggio il quale, ad ogni partita, si sposta. Se urti contro la superficie, il gioco finirà, anche se poi ricomincerà automaticamente. Se riuscirai, avrai diritto ad un nuovo posto sul quale tentare di atterrare. Questa volta il gioco sarà più difficile. Il programma continua finché riesci nel tuo intento. Il tasto "Z" è per la sinistra, il tasto "X" per la destra.

```
0 I=160:POKE650,255
1 PRINT"■■#####              #####■"
2 PRINT"■■■
```

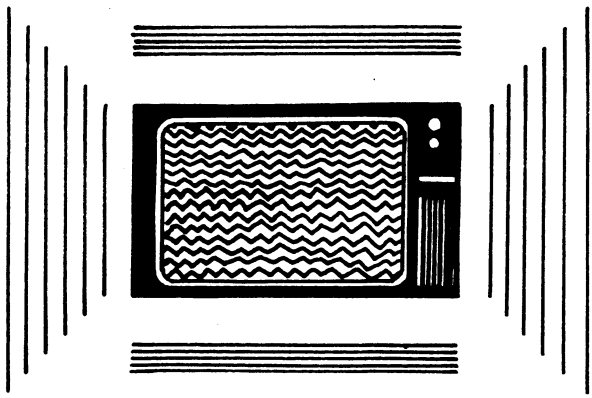
SOUND EFFECTS

Aumenta il volume della tua Tv e fatti coraggio. Il programma di Ken Staples è un vero e proprio assalto alle tue orecchie. È inoltre molto simile a uno dei programmi illustrati nel manuale delle istruzioni. Fai un confronto e nota come l'autore del programma ha migliorato e cambiato il programma.

```
1 REM**SOUND EFFECTS
2 REM**KEN STAPLES
10 S1=36874:S2=36875:S3=36876:S4=36877:V=36878:MC$="▒▒▒▒▒▒▒▒▒▒"
15 ER$="                    "
20 PRINT"▒▒▒▒▒*** SOUND EFFECTS ***"
30 PRINT"▒▒▒▒"SPC(6)"EXPLOSION":FORW=1TO1000:NEXT
40 POKEV,15:POKES4,255:FORI=15TO0STEP-.01:POKEV,I:NEXT:POKES4,0
100 PRINTMC$ER$:PRINTMC$SPC(4)"WHISTLING BOMB":FORW=1TO1000:NEXT
110 POKEV,5:FORI=230TO128STEP-1:FORJ=1TO20:NEXT:POKES3,I:NEXT:POKES3,0:POKES4,25
5
120 FORI=15TO0STEP-.05:POKEV,I:NEXT:POKES4,0
170 POKEV,15:FORI=1TO15:FORJ=200TO220+I*2:POKES3,J:NEXTJ,I:POKES3,0:POKEV,0
200 PRINTMC$ER$:PRINTMC$SPC(4)"MUSICAL NOTES":FORW=1TO1000:NEXT
210 POKEV,15:FORI=250TO200STEP-2:POKES3,I:FORJ=1TO100:NEXTJ,I:FORI=205TO250STEP2
220 POKES3,I:FORJ=1TO100:NEXTJ,I:POKES3,0:POKEV,0
400 PRINTMC$ER$:PRINTMC$SPC(8)"BEDLAM":FORW=1TO1000:NEXT
410 POKEV,15:FORI=1TO100:POKES3,INT(RND(1)*128)+128:FORJ=1TO10:NEXTJ,I:POKES3,0:
POKEV,0
500 PRINTMC$ER$:PRINTMC$SPC(6)"RED ALERT":FORW=1TO1000:NEXT
510 POKEV,15:FORI=1TO10:FORJ=180TO235STEP2:POKES3,J:FORJ1=1TO10:NEXTJ1,J:POKES3,
0
520 FORJ1=1TO100:NEXTJ1,I:POKEV,0
700 PRINTMC$ER$:PRINTMC$SPC(6)"LASER BEAM":FORW=1TO1000:NEXT
710 POKEV,15:FORI=1TO30:FORJ=250TO240STEP-1:POKES3,J:NEXTJ:FORJ=240TO250:POKES3,
J
720 NEXTJ:POKES3,0:NEXT:POKEV,0
900 PRINTMC$ER$:PRINTMC$SPC(8)"POLICE":FORW=1TO1000:NEXT
910 POKEV,15:FORI=1TO10:POKES2,235:FORJ=1TO300:NEXT:POKES2,0:POKES1,235:
920 FORJ=1TO300:NEXT:POKES1,0:NEXTI:POKEV,0
1000 PRINTMC$ER$:PRINTMC$SPC(6)"TELEPHONE":FORW=1TO1000:NEXT
1010 POKEV,15::FORI=1TO5:FORJ=1TO50:POKES3,228:FORJ1=1TO8:NEXT:POKES3,0:NEXT
1020 FORJ=1TO1500:NEXTJ,I:POKEV,0
1100 PRINTMC$ER$:PRINTMC$SPC(5)"ENGAGED TONE":FORW=1TO1000:NEXT
1110 POKEV,5:FORI=1TO15:POKES3,200:FORJ=1TO500:NEXT:POKES3,0:FORJ=1TO500:NEXTJ,I
:POKEV,0
1200 PRINTMC$ER$:PRINTMC$SPC(9)"BIRDS":FORW=1TO1000:NEXT
1205 POKEV,15:FORI=1TO40:FORJ=254TO240+INT(RND(1)*10)STEP-1:POKES3,J:NEXT:POKES3
,0
1210 FORJ=0TOINT(RND(1)*100)+120:NEXTJ,I:POKEV,0
1300 PRINTMC$ER$:PRINTMC$SPC(5)"QUATERMASS":FORW=1TO1000:NEXT
```

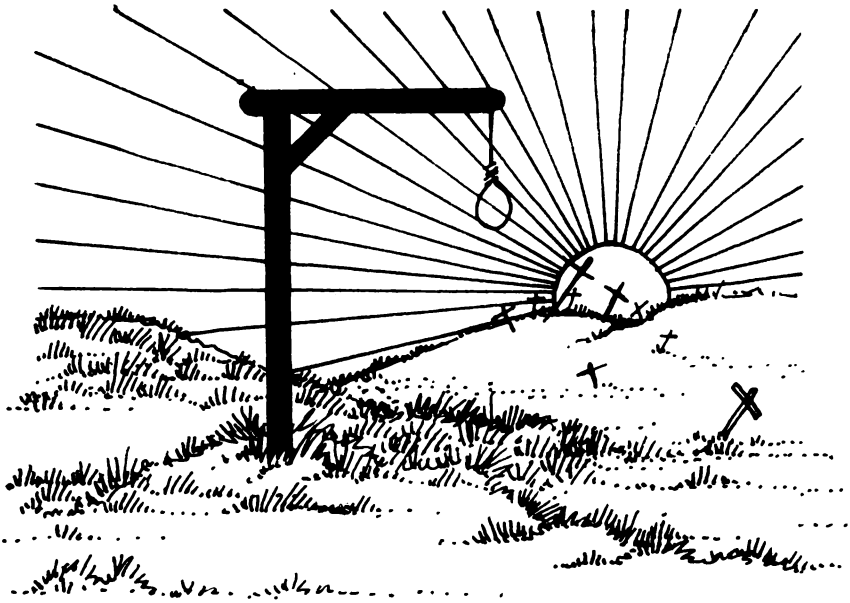
```
1310 FORI=0TO15:POKEV,I:FORJ=220TO240:POKES1,J:FORJ1=1TO10:NEXTJ1,J,I:POKES4,240
1320 FORI=1TO200:NEXT:FORI=254TO200STEP-2:POKES1,I:NEXT:FORI=15TO0STEP-.05:POKEV
,I:NEXT
1330 POKES4,0:POKES1,0
1400 PRINTMC$ER$:PRINTMC$SPC(5)"OCEAN WAVES":FORW=1TO1000:NEXT
1410 POKES4,135:FORI=1TO10:D=INT(RND(1)*90)+30:FORJ=0TO10:POKEV,J:FORJ1=1TOD:NEX
TJ1,J
1415 FORJ=15TO0STEP-1:POKEV,J:FORJ1=1TOD:NEXTJ1,J,I:POKES4,0:POKEV,0
1500 PRINTMC$ER$:PRINTMC$"VANISHING UFO":FORW=1TO1000:NEXT
1510 POKEV,15:FORI=130TO254:POKES3,I:FORJ=1TO40:NEXTJ,I:POKEV,0:POKES3,0
1600 PRINTMC$ER$:PRINTMC$SPC(5)"UFO LANDING":FORW=1TO1000:NEXT
1605 POKEV,15:FORI=1TO20:FORJ=220-ITO160-ISTEP-4:POKES3,J:NEXTJ
1610 FORJ=160-ITO220-ISTEP4:POKES3,J:NEXTJ,I:POKEV,0:POKES3,0
1700 PRINTMC$ER$:PRINTMC$SPC(5)"UFO SHOOTING":FORW=1TO1000:NEXT
1710 POKEV,15:FORI=1TO15:FORJ=200TO220+I*2:POKES3,J:NEXTJ,I:POKES3,0:POKEV,0
1800 PRINTMC$ER$:PRINTMC$SPC(5)"WOLF WHISTLE":FORW=1TO1000:NEXT
1810 POKEV,15:FORI=150TO240:POKES3,I:NEXT:POKES3,0:FORJ1=1TO200:NEXT
1820 FORI=205TO128STEP-.3:POKES3,I:NEXT:POKES3,0:POKEV,0
1900 PRINTMC$ER$:PRINTMC$SPC(6)"DANGER!!!":FORW=1TO1000:NEXT
1910 POKEV,15:FORI=1TO30:POKES1,220:FORJ=1TO100:NEXT:POKES1,0:FORJ=1TO100:NEXTJ,
I
1920 POKEV,0
2000 PRINTMC$ER$:PRINTMC$SPC(8)"BLIPS":FORW=1TO1000:NEXT
2005 FORJ=1TO20:POKEV,15:POKES3,220:FORI=1TO5:NEXT:POKES3,0:FORI=1TO500:NEXT:POK
ES3,200
2010 FORI=1TO5:NEXT:POKES3,0:FORI=1TO500:NEXTI,J:POKEV,0
3000 PRINTMC$ER$:PRINTMC$SPC(3):PRINT"LEAVING HARBOUR":FORW=1TO1000:NEXT
3010 FORI=1TO3:POKEV,15:POKES1,210:POKES4,254::FORJ=1TO1500:NEXT
3020 POKES1,0:POKES4,0:POKEV,0:FORJ=1TO500:NEXTJ,I
4000 PRINTMC$ER$:PRINTMC$SPC(5)"INTO ACTION":FORW=1TO1000:NEXT
4010 FORT=1TO2:POKEV,15:FORI=140TO220STEP5:POKES3,I::POKES4,254:FORJ=1TO5:NEXTJ,
I
4020 POKEV,0:POKES4,0:POKES3,0:FORW=1TO100:NEXTW,T:FORW=1TO2000:NEXT:A=A+1:IFA=3
THEN5000
4030 GOTO4010
5000 PRINTMC$ER$:PRINTMC$SPC(5)"*THE END*":PRINTSPC(5)::FORI=1TO11:PRINT"~"::NEX
T:PRINT
5010 FORI=1TO4000:NEXT:RUN

READY.
```



HANGMAN

Questa è una versione grafica e musicale del popolare gioco di parole. Le lettere dell'alfabeto sono esposte sulla sinistra del video e, dopo aver selezionato una lettera, questa stessa si presenta rovesciata, per evitare che tu la scelga di nuovo.
Poni una freccia sotto le lettere, spostandola come vuoi usando i tasti "K" (sinistra) e "L" (destra). Batti sulla sbarra spaziatrice quando la freccia è sotto la lettera richiesta. L'uomo impiccato è disegnato sulla destra del video. Se non indovini la parola, una scatola viene rimossa dalla parte sottostante l'uomo ed egli cade per terra. Il collo si allunga notevolmente. Viene suonato un passo della Marcia Funebre e la parola nascosta è rivelata. Se invece riesci ad indovinare prima del tuo tempo limite, sentirai un passo del Coro dell'Alleluia. L'autore di questo gioco è Mike O'Neill.

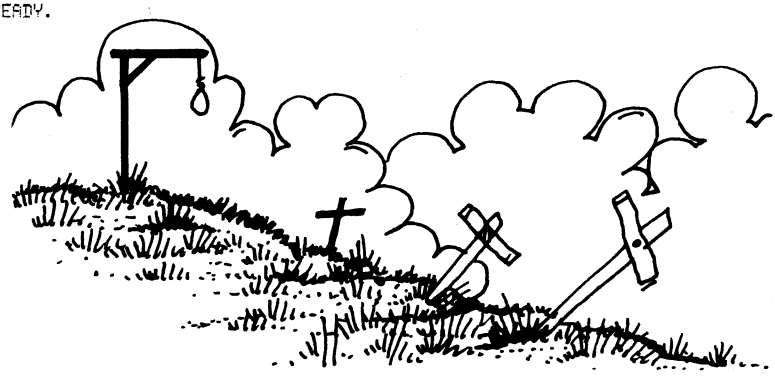


HANGMAN

```
1 GOSUB5000:POKE36879,27
10 DIMZ$(53):R=RND(-TI):S1=36874:S2=36875:V=36878
11 FORI=1TO53:READZ$(I):NEXT
12 DATATWO,SIX,BASIC,VIC,DOG,BOX,TOP,APPLE,RHYTHM,HYMN,SKY,SYRINGE,KEY,WRIST,TYR
ANT
13 DATALIGHT,HYENA,CYANIDE,ADD,DYE,FLY,BUY,GIN,MING,ARM,VASE,VET,BISCUIT,LIMP,BL
IGHT
14 DATAFRAME,FRIGHT,RING,BLOOD,SHOW,HANG,HAWK,KING,CLOCK,COMB,SLY,STICK,BLOCK,CR
AG,FLACK
15 DATAMOOSE,VELCRO,HELL,WHISKER,SHIRT,HOSE,SADDLE,GLASS
100 PRINT"■":I=INT(RND(1)*53)+1:G=0
110 A$=Z$(I)
120 L=LEN(A$):IN=0
130 FORI=1TOL:POKEI+7770,ASC(MID$(A$,I,1))-64:POKE38512+I,0:POKE7792+I,46:NEXT
200 FORI=1TO9:POKE38686+I,0:POKE7966+I,I:NEXT
210 FORI=10TO18:POKE38743+I,0:POKE8023+I,I:NEXT
220 FORI=19TO26:POKE38800+I,0:POKE8080+I,I:NEXT
300 P=7989:POKE30720+P,0:POKEP,30
310 GETA$:IFA$="K"THENGOTO350
320 IFA$="L"THENGOTO400
330 IFA$=" "THENGOTO450
340 GOTO310
350 POKEP,32:P=P-1:IFP=8054THENP=7997
360 IFP=8120THENP=8063
370 IFP=7988THENP=8128
380 POKEP+30720,0:POKEP,30:GOTO310
400 POKEP,32:P=P+1:IFP=7998THENP=8055
410 IFP=8064THENP=8121
420 IFP=8129THENP=7989
430 POKEP+30720,0:POKEP,30:GOTO310
450 IFPEEK(P-22)>128THEN310
460 POKEP-22,PEEK(P-22)+128
470 Y=0:FORI=1TOL
480 IFPEEK(7770+I)=PEEK(P-22)-128THENY=1:IN=IN+1:POKE38490+I,0
490 NEXT
500 IFY=0THEN600
510 IFIN=LTHEN7000
520 GOTO310
600 G=G+1:IFG=9THEN900
610 ONGGOTO620,640,660,680,700,720,740,760
620 PRINT"[illegible]";
630 PRINT"[illegible]";:GO
TO310
631 REMPRINT"[illegible]";:GOTO310
640 PRINT"[illegible]";:GOTO310
660 PRINT"[illegible]";:GOTO310
680 PRINT"[illegible]";:GOTO310
700 PRINT"[illegible]";:GOTO310
720 PRINT"[illegible]";:GOTO310
740 PRINT"[illegible]";:GOTO310
760 PRINT"[illegible]";:GOTO310
900 FORI=1TO3:PRINT"[illegible]";:NEXT
910 FORK=0TO44STEP22:FORJ=9TO0STEP-1:FORI=1TO3
920 M=7872+I+J*22+K:POKEM,PEEK(M-22):POKEM+30720,PEEK(M+30698)
930 NEXTI,J,K:GOTO9000
3000 GETA$:IFA$<>" "THEN3000
3010 INPUT"[illegible]ANOTHER GAME (Y/N)";A$
3020 IFA$=""THEN3010
3030 IFA$="Y"THEN100
```

```
3040 IFA$="N"THENPRINT"":END
3050 GOTO3010
5000 PRINT"        HANGMAN":PRINT"   BY MIKE O'NEILL"
5010 PRINT" HIT ANY KEY TO START"
5020 GETA$:IFA$=""THEN5020
5030 RETURN
7000 POKES1,0:GOSUB8000:GOSUB8000
7010 POKES1,225:POKES2,225:FORJ=1TO2
7020 POKEV,15:FORI=1TO80:NEXT:POKEV,0:FORI=1TO30:NEXTI
7030 POKEV,15:FORI=1TO50:NEXT:POKEV,0:FORI=1TO30:NEXTI
7040 POKEV,15:POKES1,228:POKES2,228:FORI=1TO200:NEXT:POKEV,0:FORI=1TO100:NEXTI
7050 POKEV,15:POKES1,225:POKES2,225:FORI=1TO100:NEXT:POKEV,0:FORI=1TO50:NEXTI
7060 FORI=1TO200:NEXTI,J
7100 POKEV,15:FORI=1TO200:NEXT:POKEV,0:FORI=1TO100:NEXT
7110 POKES1,228:POKES2,228:POKEV,15:FORI=1TO200:NEXT:POKEV,0:FORI=1TO100:NEXT
7120 POKEV,15:POKES1,232:FORI=1TO200:NEXT:POKEV,0:FORI=1TO100:NEXT
7130 POKEV,15:POKES1,231:POKES2,231:FORI=1TO400:NEXT:POKEV,0:FORI=1TO100:NEXT
7140 POKEV,15:POKES1,232:POKES2,232:FORI=1TO500:NEXT:POKEV,0:FORI=1TO100:NEXT
7200 GOTO3000
8000 POKEV,0:POKES1,232:POKES2,232:POKEV,15:FORI=1TO500:NEXT:POKEV,0:FORI=1TO200
:NEXT
8005 POKES1,225:POKES2,225
8010 POKEV,15:FORI=1TO200:NEXT:POKEV,0:FORI=1TO50:NEXT:POKES1,228:POKES2,228:POK
EV,15
8020 FORI=1TO200:NEXT:POKEV,0:FORI=1TO50:NEXT:POKES1,225:POKES2,225
8030 POKEV,15:FORI=1TO200:NEXT:POKEV,0:FORI=1TO500:NEXT
8100 RETURN
9000 POKEV,0:POKES1,195:POKES2,225:POKEV,15:FORI=1TO1000:NEXT
9010 POKEV,0:FORI=1TO50:NEXT:POKES1,163:POKEV,15:FORI=1TO1000:NEXT
9020 POKEV,0:FORI=1TO50:NEXT:POKES1,0:POKEV,15:FORI=1TO100:NEXT:POKEV,0:FORI=1TO
50
9030 NEXT:POKES1,195:POKEV,15:FORI=1TO1000:NEXT:POKEV,0:FORI=1TO100:NEXT
9040 POKES1,163:POKES2,229:POKEV,15:FORI=1TO1000:NEXT:POKEV,0
9050 POKES1,0:POKES2,228:POKEV,15:FORI=1TO100:NEXT:POKEV,0:FORI=1TO100:NEXT
9060 POKES1,195:POKES2,228:POKEV,15:FORI=1TO600:NEXT:POKEV,0:FORI=1TO50:NEXT
9070 POKES1,0:POKES2,225:POKEV,15:FORI=1TO500:NEXT:POKEV,0:FORI=1TO100:NEXT
9080 POKES1,163:POKEV,15:FORI=1TO700:NEXT:POKEV,0:FORI=1TO50:NEXT
9090 POKES1,0:POKEV,15:FORI=1TO200:NEXT:POKEV,0:FORI=1TO50:NEXT
9100 POKES1,195:POKEV,15:FORI=1TO1000:NEXT:POKEV,0
9110 FORI=1TOL:POKE38490+I,0:NEXT
9200 GOTO3000

READY.
```



SIMON

Questa è una versione del gioco che porta lo stesso nome. Il computer seleziona una serie di colori accompagnata da suoni, e tu devi ripetere esattamente la sequenza. Ci sono quattro zone di colori differenti, rispettivamente sulla parte superiore del video, in basso, a destra e a sinistra. La sequenza parte con un colore selezionato a caso e ogni volta che la sequenza è ripetuta viene aggiunto un altro colore. La massima lunghezza della sequenza è da cinque a 20. Per facilitare il gioco si usa più di un tasto per ogni colore.
Tasti usati:
Colore in alto — qualsiasi numero
Colore in fondo — barra spaziatrice
Colore a sinistra — qualsiasi carattere
Colore a destra — qualsiasi tasto funzionale

```
10 PRINT"□⌧        SIMON":PRINT"⌧   BY MIKE O'NEILL⌧"
20 PRINT"         KEYS⌧"
21 PRINT"      ┌──────────┐"
22 PRINT"      │    ANY   │"
23 PRINT"      │  NUMBER  │"
24 PRINT"┌──────────┬──────────┬──────────┐"
25 PRINT"□│   ANY    │   ANY    │"
26 PRINT"□│CHARACTER │FUNCTION  │"
27 PRINT"□└──────────┴──────────┴──────────┘"
28 PRINT"□      │   SPACE  │"
29 PRINT"      │    BAR   │"
30 PRINT"      └──────────┘"
35 PRINT"⌧ENTER MAXIMUM SEQUENCE"
36 INPUT"    LENGTH (5-20)";M%:IFM%<5ORM%>20THEN36
40 PRINT"HIT ANY KEY TO START"
50 GETA$:IFA$=""THEN50
60 POKE36879,25
70 S=36876:V=36878:POKEV,0:POKES,0:DIMA(20)
80 R=RND(-TI)
100 PRINT"□     ⌧              "
110 FORI=1TO6:PRINT"■     ⌧ ■           ⌧ ■":NEXT
120 PRINT"⌧                          "
130 FORI=1TO6:PRINT"□⌧ ■          ⌧  ■          ⌧ ":NEXT
140 PRINT"□⌧                          "
150 PRINT"□";:FORI=1TO6:PRINT"■     ⌧ ■           ⌧ ■":NEXT
160 PRINT"     ⌧              ⌧"
170 FORI=1TO1000:NEXT
200 N=0:Z=500
210 N=N+1
220 A(N)=INT(RND(1)*4+1)
230 FORL=1TON
240 ONA(L)GOSUB1000,2000,3000,4000
```

```
250 NEXT
300 FORL=1TON
310 C=0
320 GETA$:IFA$=""THEN320
330 B=ASC(A$)
340 IFB>47ANDB<58THENC=1:REM NUMBER
350 IFB>64ANDB<91THENC=2:REM CHARACTER
360 IFB>132ANDB<141THENC=3:REM FUNCTION
370 IFB=32THENC=4:REM SPACE
380 IFC=0THEN320
400 IFA(L)<>CTHENGOTO6000
410 ONCGOSUB1000,2000,3000,4000
420 NEXT
430 IFN=M%THEN5000
440 FORI=1TO2000:NEXT
500 GOTO210
1000 POKES,195:POKEV,15:PRINT"▒▒▒▒▒▒▒▒";:FORI=1TO6:PRINT"▒▒▒▒▒▒▒▒
```

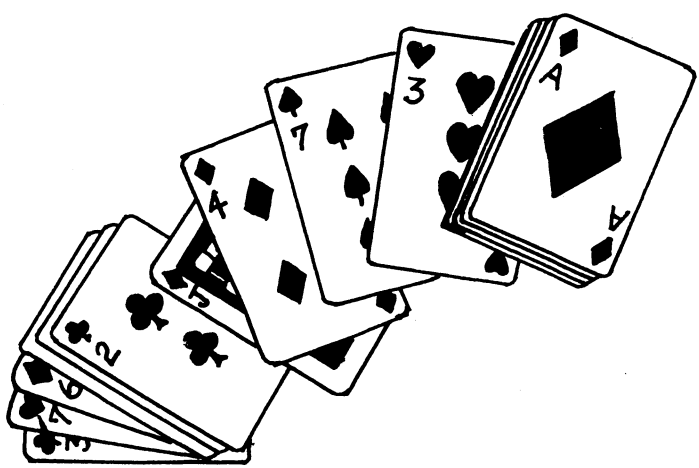
PONTOON

Preparati a perdere la camicia in questo spietato gioco. Corrisponde al nostro "Ventuno" ed è stato scritto da Ken Staples. Il computer è il banco e tu giochi sempre per primo. Questo significa che il furbo VIC deve soltanto eguagliare la tua mano e vince la tua puntata. Il vincitore di ogni partita è colui che totalizza il più alto punteggio. Le eccezioni sono le seguenti:
"VENTUNO" — Questa mano è composta da dieci carte e un asso. Vince tutte le altre mani.
"LA MANO DI CINQUE CARTE" — Cinque carte per un totale di 21 o anche meno vince su tutte le altre mani, salvo "VENTUNO".
Se il giocatore segna un punteggio superiore a 21 "sballa", e il banco vince automaticamente.

```
1 REM*PONTOON
2 REM*KEN STAPLES
5 Q=25:P=25
6 DIMD(5),S(5),H(5),C(5)
10 PRINT"■":POKE36879,29:M$="■■■■■■■■":O$="                    "
100 L=7681:C=38401:GOSUB5000:A1=E
110 INPUT"■■■■■■■BID £";A:B=FRE(0):B=A
120 IFA>5THENPRINTM$"■MAX.BID £5":FORW=1TO4000:NEXT:PRINTM$O$:GOTO110
200 L=7685:C=38405:X=1:GOSUB5000:X=0:A2=E:GOSUB2000:IFFGOTO600
300 L=7689:C=38409:GOSUB5000:A3=E:GOSUB2000:IFFGOTO600
400 L=7693:C=38413:GOSUB5000:A4=E:GOSUB2000:IFFGOTO600
500 L=7697:C=38417:GOSUB5000:A5=E:GOSUB2000
600 A1=0:A2=0:A3=0:A4=0:A5=0:M$="■■■■■■■■■■■■■■■■"
610 L=7857:C=38577:GOSUB5000:A1=E:GOSUB3000
700 L=7861:C=38581:X=1:GOSUB5000:X=0:A2=E:GOSUB3000:IFFGOTO1500
800 L=7865:C=38585:GOSUB5000:A3=E:GOSUB3000:IFFGOTO1500
900 L=7869:C=38589:GOSUB5000:A4=E:GOSUB3000:IFFGOTO1500
1000 L=7873:C=38593:GOSUB5000:A5=E:GOSUB3000
1500 IFN=1THENN=0:P=P+A:Q=Q-A
1510 IFO=1THENO=0:P=P-A:Q=Q+A
1512 IFQ<=0THENFORW=1TO1000:NEXT:PRINT"■■■CONGRATULATIONS!.YOU HAVE BROKEN THE B
ANK!":END
1515 IFP<=0THENFORW=1TO1000:NEXT:PRINT"■■■SORRY! YOU ARE BROKE!":END
1520 PRINT"■BANK £"Q:PRINT"■PLAYER £"P
1530 A=0:A1=0:A2=0:A3=0:A4=0:A5=0:I=0:J=0:K=0:M=0:PT=0:FC=0:FORW=1TO5000:NEXT:GO
TO10
2000 F=0:PRINTM$O$:IFA3>0THEN2020
2010 IFA1=1ANDA2=10ORA2=1ANDA1=10THENPRINTM$"PONTOON":F=1:PT=1:RETURN
2020 T=A1+A2+A3+A4+A5:PRINTM$"POINTS"T;
2025 IF(A1=1ORA2=1ORA3=1ORA4=1)ANDT+10<21THENPRINT"OR"T+10
2030 FORW=1TO4000:NEXT:PRINT:IFT>21THENPRINTM$O$:PRINTM$"SORRY! BUST!":F=1:RETUR
N
2036 IFA5>0ANDT<=21THENPRINTM$O$:PRINTM$"5 CARD TRICK":FC=1:RETURN
2040 PRINTM$"1 TWIST.2 BUY.3 STICK":INPUT"WHICH";R:IFR>3THENPRINTM$"■"O$:GOTO204
0
2060 PRINTM$O$:PRINT"■■■■■■■■■"O$:ONRGOTO2110,2120
2080 IFA1=1ORA2=1ORA3=1ORA4=1ORA5=1THENT=T+10:IFT>21THENT=T-10
2090 IFT<15THENPRINTM$"■CANNOT STICK UNDER 15":FORW=1TO4000:NEXT:GOTO2040
2100 PRINTM$"STUCK ON"T"STAKE £"A:F=1:RETURN
2110 IFT<11THENPRINTM$"■CANNOT TWIST UNDER 11":FORW=1TO4000:NEXT:GOTO2040
2115 RETURN
2120 PRINTM$O$:PRINTM$"STAKE £"A;:INPUT"BID £";G:IFG<=BTHENA=A+G:RETURN
2140 PRINTM$O$:PRINTM$"■ONLY UP TO FIRST BID":FORW=1TO4000:NEXT:GOTO2120
3000 U=A1+A2+A3+A4+A5:PRINTM$O$
3010 IFA1=10ANDA2=1ORA1=1ANDA2=10THENPRINTM$"PONTOON.BANK WINS":F=1:O=1:RETURN
3012 PRINTM$"POINTS"U;
3015 IF(A1=1ORA2=1ORA3=1ORA4=1)ANDU+10<=21THENPRINT"OR"U+10:IFU+10>=TTHEN3090
3020 FORW=1TO4000:NEXT:IFPT=1ORU>21ANDT<=21THENPRINTM$"■YOU WIN!":F=1:N=1:RETURN
3025 IFT>21ANDA2>0ORA5>0ANDU<=21THENPRINTM$"■■BANK WINS!":F=1:O=1:RETURN
3070 IFFC=0ANDU<=21ANDU>=TTHENPRINTM$"■■BANK WINS!":F=1:O=1:RETURN
3080 RETURN
3090 PRINT"■■BANK WINS!":F=1:O=1:RETURN
5000 POKEL,79:POKEC,0:POKEL+1,99:POKEC+1,0:POKEL+2,80:POKEC+2,0
5010 POKEL+22,101:POKEC+22,0:POKEL+24,103:POKEC+24,0
5020 POKEL+44,101:POKEC+44,0:POKEL+46,103:POKEC+46,0
5030 POKEL+66,101:POKEC+66,0:POKEL+68,103:POKEC+68,0
5040 POKEL+88,76:POKEC+88,0:POKEL+89,100:POKEC+89,0:POKEL+90,122:POKEC+90,0
6000 S=INT(RND(1)*4)+1:IFS=1THENS=65:H=0:I=I+1:GOSUB7000:GOSUB8000:S(I)=D:RETURN
6020 IFS=2THENS=83:H=2:J=J+1:GOSUB7000:GOSUB9000:H(J)=D:RETURN
6030 IFS=3THENS=88:H=0:K=K+1:GOSUB7000:GOSUB10000:C(K)=D:RETURN
```

```
6040 IFS=4THENS=90:H=2:M=M+1:GOSUB7000:GOSUB11000:D(M)=D:RETURN
7000 D=INT(RND(1)*13)+1:IFD=1THENE=1
7020 IFD=2THENE=2:D=50
7030 IFD=3THENE=3:D=51
7040 IFD=4THENE=4:D=52
7050 IFD=5THENE=5:D=53
7060 IFD=6THENE=6:D=54
7070 IFD=7THENE=7:D=55
7080 IFD=8THENE=8:D=56
7090 IFD=9THENE=9:D=57
7100 IFD=10THENE=10:D=0
7110 IFD=11THENE=10:D=10
7120 IFD=12THENE=10:D=17
7130 IFD=13THENE=10:D=11
7140 IFA1=1ANDX=1ANDE=1THEN7000
7170 RETURN
8000 FORW=1TOI:IFD<>S(W)THEN8030
8020 F=1
8030 NEXT:IFFTHENF=0:GOSUB7000:GOTO8000
8040 GOSUB12000:RETURN
9000 FORW=1TOJ:IFD<>H(W)THEN9030
9020 F=1
9030 NEXT:IFFTHENF=0:GOSUB7000:GOTO9000
9040 GOSUB12000:RETURN
10000 FORW=1TOK:IFD<>C(W)THEN10030
10020 F=1
10030 NEXT:IFFTHENF=0:GOSUB7000:GOTO10000
10040 GOSUB12000:RETURN
11000 FORW=1TOM:IFD<>D(W)THEN11030
11020 F=1
11030 NEXT:IFFTHENF=0:GOSUB7000:GOTO11000
11040 GOSUB12000:RETURN
12000 POKEL+23,S:POKEC+23,H
12010 IFD=0THENPOKEL+45,49:POKEC+45,0:POKEL+46,48:POKEC+46,0:RETURN
12020 POKEL+45,D:POKEC+45,0
12030 RETURN

READY.
```

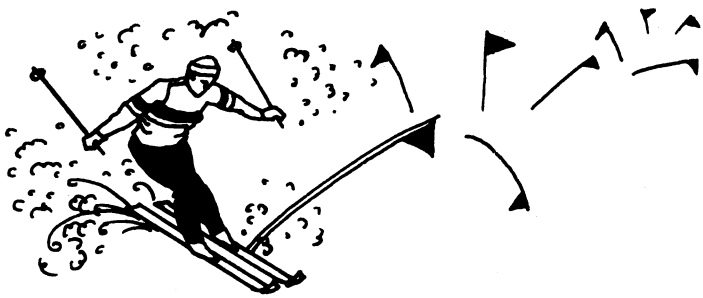


SKY RUN

In questo gioco d'azione scritto da Kevin Smith, controlli uno sciatore che devi guidare attraverso 20 valichi, con un qualsiasi livello di abilità. Vanno da uno (professionista) a quattro (principianti).
Il tasto "Z" sposta lo sciatore a sinistra, "M" a destra. Il gioco ricomincia automaticamente dopo uno scontro.

```
5 PRINT"▒":POKE650,255
10 PRINT"   SKI RUN"
15 B=7691
20 PRINT"▒Z=LEFT,M=RIGHT"
25 PRINT"▒SKI THROUGH THE GATES"
30 PRINT"▒SPEED:▒1=PRO"
40 PRINT"4=BEGINNER"
45 POKE198,0
50 INPUTA:A=A*15
60 PRINT"▒"
100 REM ROUTINE
110 PRINT"▌                    ▌"
120 POKEB,25
130 C=C+1:IFC=24 THEN200
135 POKEB,96
140 GETA$
150 IFA$="Z" THENB=B-1
160 IFA$="M" THENB=B+1
165 POKEB,25
170 FORT=1TOA:NEXT
180 GOTO100
200 C=0:D=INT(RND(1)*17)
205 E=E+1:IFE=21 THENPRINT"▒  GAME  OVER":FORC=1TO2000:NEXT:RUN
210 PRINTTAB(D)"I   I"
220 GOTO100

READY.
```



DIAMOND GLORY

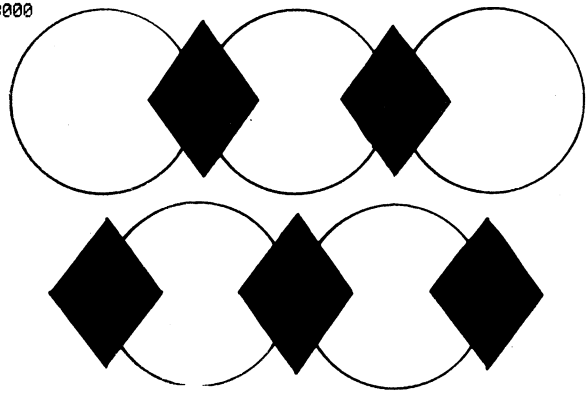
Questo gioco "a labirinto" è un tipo di "solitario". Può durare alcuni secondi come alcune ore. Tu controlli da uno a cinque pezzi a forma di cerchio; lo stesso numero di pezzi romboidali restano immobili durante la partita. L'obiettivo è di spostare i tuoi pezzi verso i rombi in modo che ognuno copra un rombo diverso. Può sembrare facile, ma aspetta a giocare con quattro o cinque pezzi.

"A" — verso l'alto
"S" — verso il basso
"K" — verso sinistra
"L" — verso destra
"SPACE" — per fermarti

```
1 REM DIAMOND GLORY
10 R=RND(-TI):S=36876:V=36878:DIMP(5),C(5)
20 PRINT"▯▯▯ DIAMOND GLORY ":PRINT"▯▯   BY MIKE O'NEILL":POKEV,0
22 PRINT"▯ KEYS:":PRINT"▯ A=UP":PRINT" S=DOWN":PRINT" K=LEFT":PRINT" L=RIGHT"
24 PRINT" SPACE=STOP▯▯"
30 INPUT" ENTER LEVEL (1-5)";R:IFR>5ORR<1THEN30
40 IR=800-R*100:PRINT
50 INPUT" HOW MANY (1-5)";NN:NN=INT(NN):IFNN>5ORNN<1THEN50
60 POKE36879,57:PRINT"■"
100 PRINT"▯▯┌──────────────────────┐
101 PRINT"▯|                      |
102 PRINT"▯| ┌──── ──┬──────┐ | |"
103 PRINT"▯| |        |     | | |"
104 PRINT"▯| | ┌──────┴──┐ | | |"
105 PRINT"▯|   |         | | | |"
107 PRINT"▯| | | ┌──── ┐   | | |"
108 PRINT"▯| | | |      | | | ├─┤"
109 PRINT"▯| | |   ┌ ──┘ |   | |"
110 PRINT"▯| |   | |      | |   |"
111 PRINT"▯| | | | └──── | | | |"
112 PRINT"▯| ├─┤ |        | └─┘ |"
113 PRINT"▯|   ├─┤ ├─ ──┘      |"
114 PRINT"▯| |   | |        ┌─┐ |"
115 PRINT"▯| | | | └──── ┐ | | |"
116 PRINT"▯├ ┤ | |        | | | |"
117 PRINT"▯| └─┤   ┌ ──────┘ | | |"
118 PRINT"▯|   | | |        | ├ ┤"
119 PRINT"▯| ├ ┘   ├──┐ ┌ ─┤   |"
120 PRINT"▯|     |     |      | |"
121 PRINT"▯└─┴──┴─┴──┴─┴──┴─┴─┴─┘▯"
190 FORN=1TONN
```

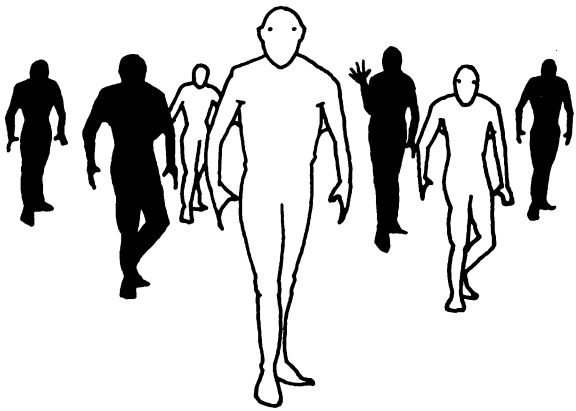
```
200 P(N)=INT(RND(1)*418+7724):IFPEEK(P(N))<>32THEN200
210 POKEP(N)+30720,1:POKES,200:FORJ=1TO10
211 POKEV,0:POKEP(N),32:FORI=1TO50:NEXT:POKEP(N),81:POKEV,15:FORI=1TO50:NEXTI,J:
POKEV,0
212 C(N)=32
220 NEXT
222 FORN=1TONN
224 CC=INT(RND(1)*418+7724):IFPEEK(CC)<>32THEN224
226 POKECC+30720,2:POKES,200:FORJ=1TO10
228 POKEV,0:POKECC,32:FORI=1TO50:NEXT:POKECC,90:POKEV,15:FORI=1TO50:NEXTI,J:POKE
V,0
230 NEXT
300 A$=""
310 GETB$:IFB$=""THEN330
320 A$=B$:GOTO310
330 IFA$=""ANDD=0THEN405
335 IFA$=""THEN400
340 IFA$="A"THEND=-22
350 IFA$="S"THEND=22
360 IFA$="K"THEND=-1
370 IFA$="L"THEND=1
380 IFA$=" "THEND=0:GOTO405
400 POKEV,15:FORI=1TO100:NEXT:POKEV,0
405 FORN=1TONN
410 CH=PEEK(P(N)+D):IFCH=32THEN420
412 IFCH<>90THEN450
414 GOTO2000
420 POKEP(N),C(N):P(N)=P(N)+D:C(N)=PEEK(P(N)):CL(N)=PEEK(P(N)+30720)
430 POKEP(N)+30720,CL(N):POKEP(N),81
450 NEXT
460 FORI=1TOIR:NEXT
500 GOTO300
2000 J=0:FORI=1TONN
2010 IFC(I)<>90ANDI<>NTHENJ=1
2020 NEXT
2030 IFJ=1THEN420
2040 POKEP(N),32:P(N)=P(N)+D
2100 POKES,0:FORI=1TO100STEP3:POKES,150+I:POKEV,15:FORK=1TO5:NEXT
2110 FORJ=1TONN:POKEP(J),32:NEXT:POKEV,0:FORK=1TO5:NEXT
2120 FORJ=1TONN:POKEP(J),81:NEXTJ,I
3000 INPUT"[illegible]ANOTHER GAME(Y/N)";A$:IFA$="N"THENPOKE36879,27:END
3010 IFA$="Y"THEN30
3020 GOTO3000

READY.
```



ZOMBIES

Mike O'Neill ti spedisce nel bel mezzo di un'isola, circondata da "zombies" e da trappole. L'obiettivo del gioco è quello di attirare gli zombies nelle trappole, mentre cerchi di non essere mangiato da uno zombies o di non cadere tu stesso in una trappola.
Puoi spostarti orizzontalmente e verticalmente. Continuerai a muoverti nella stessa direzione finché non ne scegli una nuova. Se lo desideri, puoi restare immobile. Gli zombies si muovono sempre verso di te, senza far caso a ciò che può esserci sulla loro strada.
All'inizio del gioco hai quattro vite e ci sono 20 zombies da distruggere. Ottieni un punto per ogni zombies che cade dentro una trappola. Ogni volta che cancelli l'isola degli zombies, ottieni un premio di dieci punti, e l'isola risorge di nuovo con due zombies in più. Ciò continua fino ad un massimo di 30 zombies. Se cadi dentro una trappola o sei mangiato da uno zombies, perdi una vita e l'isola viene disegnata di nuovo. Questa volta ci saranno 20 nuovi zombies!
I tuoi controlli:
"A" — in alto
"S" — in basso
"K" — a sinistra
"L" — a destra
"S" — stop

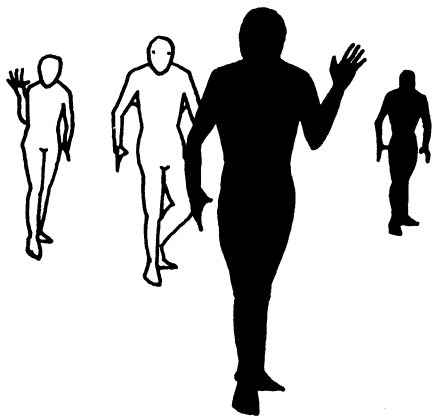


```
0 GOTO9000
1 DIMZ%(30,1)
2 I0=7746:S2=36875:S4=36877:V=36878:IS=0:POKE36879,94:IL=20
3 IN=4
10 POKEV,0:POKES2,0:GOSUB1000:GOSUB2000:GOSUB3000:GOSUB4000:GOSUB5000
20 IW=100:IC=0
100 FORI=1TOZZ
120 IFZ%(I,0)=0THEN300
130 IC=IC+1:IFIC>CZTHENIC=0
135 IFIC=0THENGOSUB500
140 ZX=-1:ZY=-22
150 IFIX>Z%(I,0)THENZX=1
160 IFIY>Z%(I,1)THENZY=22
163 IFIX=Z%(I,0)THENZX=0
165 IFIY=Z%(I,1)THENZY=0
170 T=I0+Z%(I,0)+Z%(I,1)*22+ZX+ZY:T1=PEEK(T)
190 IFT1=32THEN225
200 IFT1=81THEN6000
210 IFT1=26THEN300
220 IFT1=87THEN310
225 FORTT=1TOIW:NEXT
250 POKE(I0+Z%(I,0)+Z%(I,1)*22),32
251 POKES2,250-IZ*2 :POKEV,15
252 FORT1=1TO50:NEXT
260 POKEV,0:POKET,26:POKET+30720,0
270 Z%(I,0)=Z%(I,0)+ZX:Z%(I,1)=Z%(I,1)+ZY/22
300 NEXT
305 GOTO100
310 T=Z%(I,0)+Z%(I,1)*22+I0
320 POKET,32
321 POKEV,15
322 FORG=180TO250:POKES2,G:NEXT
323 POKEV,0:Z%(I,0)=0:IZ=IZ-1
350 CZ=IZ/6+1:IFIZ<3THENCZ=0
360 IS=IS+1:IW=IW-3
370 PRINT"SCORE";IS
380 IFIZ=0THEN8000
390 GOTO300
500 GOSUB5000
520 IFA$=""THEN650
530 IFA$="K"THEN610
540 IFA$="L"THEN620
550 IFA$="A"THEN630
560 IFA$="S"THEN640
565 IFA$=" "THEN645
600 GOTO650
610 PP=-1:GOTO650
620 PP=1:GOTO650
630 PP=-22:GOTO650
640 PP=22:GOTO650
645 PP=0:GOTO650
650 T=IP+PP:GOTO700
700 T1=PEEK(T):IFT=32THEN730
710 IFT1=26THEN6500
720 IFT1=87THEN7000
730 POKEIP,32
740 IP=T:POKET+30720,0:POKET,81
741 IFPP=-1THENIX=IX-1
742 IFPP=1THENIX=IX+1
743 IFPP=22THENIY=IY+1
```

```
744 IFPP=-22THENIY=IY-1
750 RETURN
1000 PRINT"▯▯SCORE";IS:PRINT"▯▯oooooooooooooooooooooo"
1020 FORI=1TO16
1030 PRINT"▯o                    o"
1040 NEXT
1050 PRINT"▯o                    o"
1060 PRINT"▯o                    o▯▯▯▯"
1070 PRINT"▯▯▯oooooooooooooooooooooo▯"
1090 PRINT"▯"
1091 POKE8185,87:POKE38905,1:PRINT"■"
1100 FORI=1TOIN:POKE7719+I,81:POKE38439+I,0:NEXT
1200 RETURN
2000 IZ=0:R=0.03
2020 FORI=1TO20
2030 FORT=1TO18
2032 IFIZ=30THEN2090
2035 IFIZ=ILTHEN2090
2040 IFRND(1)>RTHEN2080
2050 K=I0+I+T*22:IFPEEK(K)=26THEN2080
2060 IZ=IZ+1:Z%(IZ,0)=I:Z%(IZ,1)=T
2070 POKEK+30720,0:POKEK,26
2080 NEXTT,I
2085 GOTO2020
2090 ZZ=IZ:CZ=1+IZ/6:RETURN
3000 FORI=1TO20
3020 FORT=1TO18
3025 IFRND(1)>0.1THEN3060
3030 K=I0+I+T*22
3040 IFPEEK(K)=26THEN3060
3050 POKEK,87:POKEK+30720,1
3060 NEXTT,I:RETURN
4000 IX=INT(RND(1)*15+3)
4030 IY=INT(RND(1)*14+3)
4050 FORI=IX-1TOIX+1:FORJ=IY-1TOIY+1
4060 IP=I0+I+J*22
4070 IFPEEK(IP)<>32THEN4000
4080 NEXTJ,I
4090 IP=I0+IX+IY*22
4130 POKEIP,81:POKEIP+30720,0:PP=0
4140 POKES2,240
4200 FORI=1TO10:FORT=1TO20:NEXT
4205 POKEV,I
4210 POKEIN+7719,81:POKEIP,32:FORT=1TO20:NEXT
4215 POKEV,0
4220 POKEIN+7719,32:POKEIP,81:NEXT
4500 RETURN
5000 A$=""
5010 GETB$:IFB$=""THEN5030
5020 A$=B$:GOTO5010
5030 RETURN
6000 T=I0+Z%(I,0)+Z%(I,1)*22
6005 POKEV,15
6010 FORI=2TO30STEP2
6020 FORG=1TO10:NEXT
6030 POKET,26:POKEIP,81:POKES2,220+I
6040 FORG=1TO10:NEXT
6050 POKET,32:POKEIP,26:POKES2,200+I
6100 NEXT
6400 GOTO7300
```

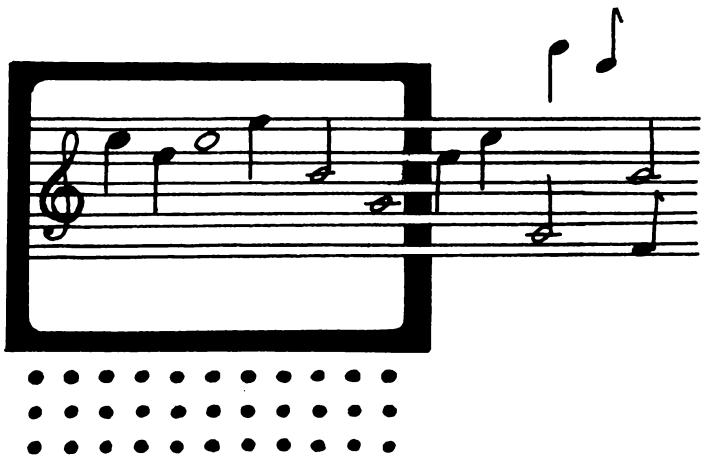
```
6500 T1=26:IC=0
6600 GOTO7050
7000 T1=87:IC=1
7050 POKEIP,32
7100 POKEV,15
7200 FORI=2TO30STEP2
7210 POKET,81:POKET+30720,0:POKES2,180+I
7220 FORG=1TO10:NEXT
7230 POKET,T1:POKET+30720,IC:POKES2,200+I
7240 FORG=1TO10:NEXT
7250 NEXT
7300 IL=20:IN=IN-1
7400 IFIN=0THEN9500
7500 GOTO10
8000 FORI=1TO10:POKEV,15
8001 FORG=1TO10
8002 POKES2,140+G+I*10
8003 NEXT
8004 PRINT"▒          BONUS"
8005 FORG=1TO10
8006 POKES2,130+G+I*10
8007 NEXT
8010 PRINT"▒                "
8015 NEXT:IS=IS+10:POKEV,0
8020 IL=IL+2:IFIL>30THENIL=30
8500 GOTO10
9000 PRINT"▒▒       ZOMBIES":PRINT"▒▒   BY MIKE O'NEILL"
9005 PRINT"▒▒ KEYS :":PRINT"▒ A=UP":PRINT" S=DOWN":PRINT" K=LEFT":PRINT" L=RIGHT
"
9006 PRINT" SPACE=STOP"
9007 PRINT"▒▒▒ HIT ANY KEY TO START"
9100 GETA$:IFA$=""THEN9100
9200 GOTO1
9500 GETA$:IFA$<>""THEN9500
9501 POKEV,0:INPUT"▒▒▒ANOTHER GAME (Y/N)   ▮▮▮";A$:IFA$="N"THENPRINT"▒"
9502 IFA$="N"THENPRINT"▒":POKE36879,27:END
9505 IFA$<>"Y"THEN9501
9600 CLR:GOTO1

READY.
```



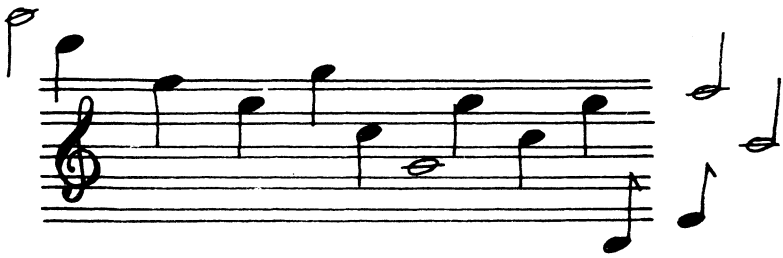
MICROMUSICA

Questo programma di musica può essere usato in due modi:
1) per ripetere un semplice insieme di suoni, come i rulli di un tamburo, allo scopo di dare un accompagnamento ritmico ad altra musica
2) per produrre melodie complete.
La musica è registrata in passi. Ogni passo consiste in note immagazzinate nei generatori bassi, medi ed alti. Il volume e la durata della nota sono memorizzati. La gamma delle note è da 128 a 255, come illustrato nel tuo libretto d'istruzioni. La gamma del volume è da 0 a 15. La durata ha il valore base di uno, così le mezze note sono registrate come 0,5, i terzi di nota come 0,333 e i quarti di nota come 0,25. Quando il programma è in funzione, inserisci i sei fattori elencati. La melodia inizierà a sentirsi quando inserisci "1" per la nota "bassa" a qualsiasi livello.
Quando "Tempo?" appare, prova prima con un valore di 100. Mentre ascolti la musica, se premi F5 potrai alterare il tempo per adattarlo alla melodia. La gamma di valore dovrebbe essere tra 1 a 1000. Se premi F7 qualsiasi numero di livello potrà essere reinserito.



```
100 PRINT"■"
110 POKE36879,136
120 GOTO340
130 REM
140 IFA(0,C)<0THENC=0
150 PRINT"□       ";C
160 FORX=36874TO36878
170 POKEX,A(X-36874,C)
180 NEXT
190 FORX=1TO(1000-S)*A(5,C)/5:NEXT
200 IFCC=0THENPOKE36878,0
210 FORX=1TO(20-(S/50))*A(5,C)/5:NEXT
220 GETA$
230 IFA$="■"GOTO280
240 IFA$="■"GOTO290
250 IFA$="■"THENCC=ABS(CC-1)
260 C=C+1:GETA$:IFA$="■"THENC=0
270 GOTO130
280 INPUT"□NEW TEMPO";S:GOTO130
290 INPUT"□STEP NO.";X
300 INPUT"LOW";A(0,X):IFA(0,X)<0GOTO130
310 INPUT"MIDDLE";A(1,X):INPUT"HIGH";A(2,X):INPUT"NOISE";A(3,X)
320 INPUT"VOLUME";A(4,X):INPUT"DURATION";A(5,X)
330 GOTO130
340 REM
350 INPUT"□TEMPO";S
360 DIMA(5,75)
370 PRINT"□";
380 X=0
390 PRINT"■ STEP NO. ";X;"■"
400 INPUT"LOW";A(0,X):IFA(0,X)<0GOTO130
410 INPUT"MIDDLE";A(1,X):INPUT"HIGH";A(2,X):INPUT"NOISE";A(3,X)
420 INPUT"VOLUME";A(4,X):INPUT"DURATION";A(5,X)
430 IFX<100THENX=X+1
440 GOTO390

READY.
```



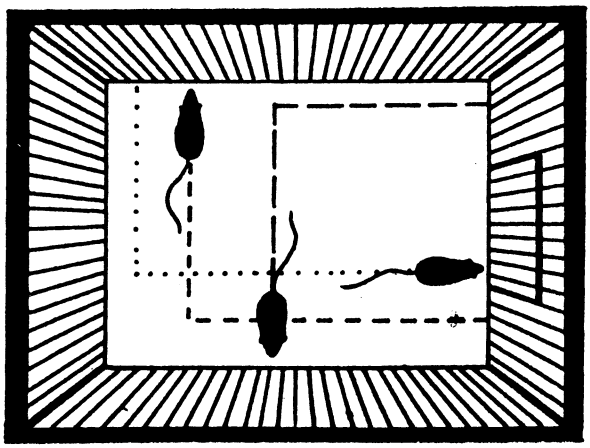
RAT TRAP

L'autore di questo programma è Mike O'Neill ed è un gioco per due. Si tratta di spostarsi per il video lasciando delle tracce dietro di sé. Lo scopo del gioco è quello di evitare la collisione con qualsiasi cosa, compresi i bordi del video e le tracce. La direzione dello spostamento non cambia finché non si sceglie una nuova direzione. Il giocatore numero 1 ha i seguenti controlli:
"A" = sopra, "S" = sotto, "D" = sinistra, "F" = destra. E il giocatore numero due: F5 = sopra, F7 = sotto, "." = sinistra, "/" = destra.

```
0 REM RAT TRAP
1 GOSUB1000:GOSUB2000
2 DIMA%(150),B%(150):V=36878
5 S=36875:POKE36879,120:C=30720
6 I%=1:POKEV,0:POKES,0
10 FORI=1TO150:A%(I)=0:B%(I)=0:NEXT
15 D1%=1:D2%=-1
100 PRINT"▒";B$,C$:PRINTS1,S2
110 PRINT"▒▒ ▒                    "
120 FORI=1TO16
130 PRINT" ▒ ▒                  ▒ "
140 NEXT
150 PRINT" ▒                    ▒"
160 A%(1)=7972:B%(1)=7981
170 POKEA%(1)+C,2:POKEA%(1),81
180 POKEB%(1)+C,5:POKEB%(1),90
190 FORI=1TO2000:NEXT
200 I%=I%+1
201 POKES,0:POKEV,15:FORI=1TO20:POKES,160+I%/2+I:NEXT
210 GETA$:IFA$=""THEN300
220 IFA$="▒"THEND2%=-22
230 IFA$="▒"THEND2%=22
240 IFA$="."THEND2%=-1
250 IFA$="/"THEND2%=1
260 IFA$="A"THEND1%=-22
270 IFA$="S"THEND1%=22
280 IFA$="D"THEND1%=-1
290 IFA$="F"THEND1%=1
295 GOTO210
300 A%(I%)=A%(I%-1)+D1%
310 IFPEEK(A%(I%))<>32THEN400
320 POKEA%(I%)+C,2:POKEA%(I%),81
350 B%(I%)=B%(I%-1)+D2%
360 IFPEEK(B%(I%))<>32THEN500
370 POKEB%(I%)+C,5:POKEB%(I%),90
371 POKES,0:POKEV,15:FORI=1TO20:POKES,160+I%/2-I:NEXT
380 FORI=1TO500:NEXT
390 GOTO200
```

```
400 S2=S2+1
410 I%=I%-1:IFI%=0THEN6
420 POKEA%(I%),32
421 POKES,250-I%/2:POKEV,15
430 FORI=1TO5:POKES,250+I-I%/2:NEXT
440 GOTO410
500 S1=S1+1
510 I%=I%-1:IFI%=0THEN6
520 POKEB%(I%),32
521 POKES,250-I%/2:POKEV,15
530 FORI=1TO5:POKES,250+I-I%/2:NEXT
540 GOTO510
1000 PRINT"▒▒      RAT-TRAP"
1005 PRINT"▒   BY MIKE O'NEILL"
1010 PRINT"▒▒▒          KEYS"
1015 PRINT"▒▒ PLAYER 1    PLAYER 2"
1017 PRINT"▒ ●●●●●●●●    ■●●●●●●●●■"
1020 PRINT"▒▒ A=UP        F5=UP"
1025 PRINT" S=DOWN      F7=DOWN"
1030 PRINT" D=LEFT      .=LEFT"
1035 PRINT" F=RIGHT     /=RIGHT"
1040 PRINT"▒▒ HIT ANY KEY TO START"
1045 GETA$:IFA$=""THEN1045
1050 RETURN
2000 PRINT"▒▒▒  ENTER YOUR NAMES"
2005 PRINT" (MAX. 9 CHARACTERS)"
2010 INPUT"▒▒PLAYER 1";B$
2020 IFLEN(B$)<10THEN2050
2030 PRINT"▒MAXIMUM 9 CHARACTERS"
2035 INPUT"PLEASE RE-ENTER";B$:GOTO2020
2050 INPUT"▒▒PLAYER 2";C$
2060 IFLEN(C$)<10THEN2090
2070 PRINT"▒MAXIMUM 9 CHARACTERS"
2075 INPUT"PLEASE RE-ENTER";C$:GOTO2060
2090 RETURN

READY.
```



JUPITER RUN

In questo programma una fila infinita di astronavi interplanetarie e di frammenti che si schiantano contro di te. Ciò che devi fare, naturalmente, è di distruggere questi oggetti ed allo stesso tempo evitare una collisione. In questo gioco hai solo una possibilità.
La tua astronave è armata con un lancia-laser che può solo sparare un colpo alla volta. Per rendere le cose ancora più difficili, inoltre, la tua arma ha una portata limitata e non può raggiungere il bersaglio sulla parte superiore del video. Questo gioco si basa su due fattori: la sopravvivenza e la resistenza, dato che devi combattere contro il costante assalto del nemico e devi tenere in considerazione la tua perdita di energia. Fai molta attenzione quando prendi la mira poiché ogni colpo del lancia-laser costa 10 unità di energia, e se il tuo livello di energia scende a zero sei considerato morto. Se il colpo va a segno guadagni 10 unità di energia con le quali puoi continuare a combattere.
Hai cinque tasti di controllo a tua disposizione:

"1" — Inclina l'astronave ad un angolo di 45° gradi verso la sinistra
"2" — Inclina l'astronave ad un angolo di 45° gradi verso destra.
"8" — Sposta l'astronave verso sinistra
"9" — Sposta l'astronave verso destra
"Ø"— Fa sparare il tuo lancia-laser

Il tuo punteggio è segnato sulla parte superiore a sinistra mentre il tuo livello di energia è visibile in alto a destra. Premi "Y" per incominciare un nuova partita.

```
10 REM JUPITER RUN
20 GOSUB1000
30 IFPEEK(653)THEN30
40 GOSUB500
80 K=PEEK(197)
90 IFK=59ANDFNT(X)>1THENX=X-1
100 IFK=4ANDFNT(X)<21THENX=X+1
110 IFK=0ANDS>1THENS=S-1
120 IFK=56ANDS<3THENS=S+1
140 IFPEEK(X)>32THEN2000
150 IFX<>OXORPEEK(X)=32THENPOKEOX,32:POKEX,S+26:OX=X
160 IFK=60THENGOSUB300
170 PRINT"[illegible]&"SC;TAB(10)"("E"[illegible]  "
180 IFE<50THENPOKE36879,10
```

```
190 IFE>50THENPOKE36879,8
200 GOTO30
300 REM FIRING
310 N=230:E=E-5:SF=SF+1
320 M=M(S):D=D(S):Y=X
330 FORI=1TO6:IFY<XTHENPOKEY,32
340 Y=Y-D
350 N=N-5:POKES3,N
360 IFPEEK(Y)>=34THEN400
370 POKEY,M:POKEFNC(Y),2
375 B=FNT(Y):IFB=0ORB=22THEN390
380 NEXT
390 POKES3,0:POKEY,32:RETURN
400 REM  HIT!
410 POKES4,0:POKES1,245:POKES2,225:POKES3,196:POKEY,39:POKEFNC(Y),1
420 SC=SC+10:E=E+10:H=H+1
470 FORI=1TO50:NEXT:POKEY,32
480 POKES1,0:POKES2,0:POKES3,0
490 RETURN
500 REM PLANT ASTEROID
510 FORI=1TOA
520 POKE646,FNR(7)+1
530 PRINT"[illegible]"TAB(FNR(20))CHR$(FNR(3)+34)
540 NEXT
550 PRINT"[illegible]":POKE218,158:
560 IFPEEK(X)>34THEN2000
570 POKEX,S+26:POKEFNC(C),1
580 RETURN
999 END
1000 REM READY
1010 PRINT"[illegible]":POKE52,28:POKE56,28:POKE55,0:POKE51,0
1020 FORI=27*8TO40*8+7
1030 READA:POKEI+7168,A:NEXT
1040 DATA136,68,42,17,160,64,32,16
1050 DATA16,16,146,146,254,130,130,0
1060 DATA17,34,84,136,5,2,4,8
1070 DATA0,0,0,16,8,4,0,0,0,0,8,16,32,0,0
1080 DATA0,0,0,0,0,0,0,0,0,0,16,16,16,0,0
1090 DATA24,112,60,254,126,124,63,0,0,0,48,124,62,24,0,0
1100 DATA0,0,56,124,82,252,120,48
1110 DATA0,0,219,146,210,82,219,0,0,0,179,170,179,170,171,0
1120 DATA0,68,16,56,16,68,0,0
1130 DATA126,64,64,120,64,64,126,0
1150 FORI=48*8TO57*8+7:POKE7168+I,PEEK(32768+I):NEXT
1200 POKE36879,8:POKE36878,15:POKE36869,255
1210 DEF FNR(X)=INT(RND(1)*X)
1220 DEF FNT(X)=22*((X-7680)/22-INT((X-7680)/22))
1230 DEF FNC(X)=X+30720
1240 X=8174
1250 S=2
1260 S1=36874:S2=S1+1:S3=S2+1:S4=S3+1
1270 FORI=1TO3:READM(I),D(I):NEXT
1280 DATA30,23,33,22,31,21
1290 A=1
1300 E=100
1999 RETURN
2000 REM END
2010 POKES4+2,42
2020 POKES1,180
2030 POKEX,39:POKEFNC(X),1
```

```
2040 FORI=1TO1000:NEXT
2050 POKES1,0:POKE198,0
2070 WAIT198,1:POKE198,0
2080 POKE36869,242
2090 PRINT"[redacted] IOU'RE DEAD!"
2100 PRINT"[redacted]ND ONLY SCOPED[redacted]",SC
2110 PRINT"[redacted]HOTS FIRED"TAB(11)SF
2120 PRINT" IITS"TAB(11)H
2130 IFH>0THENPRINT"% IITS"TAB(11)INT(H/SF*100)"[redacted]%"
2150 PRINT"[redacted]DANT TO DO BETTER?"
2160 GETA$:IFA$="Y"THENRUN
2170 IFA$="N"THENPRINT"[redacted]":POKE631,ASC("[redacted]"):POKE198,1:END
2180 GOTO2160

READY.
```

VOGON

In questo programma la tua missione è quella di distruggere una città "indifesa". Se non ti rimorde la coscienza per la descrizione del bersaglio, appena il tuo aeroplano attraversa il video spostandosi da sinistra a destra, lancia i missili premendo il tasto delle maiuscole. Se tieni questo tasto abbassato, i missili continueranno a volare invece di cadere immediatamente. Ciò ti permetterà di prendere la mira con maggiore attenzione. A mano a mano che il tempo passa, il tuo apparecchio perderà gradualmente quota. Se riesci a distruggere tutti i palazzi in basso prima che l'aereo tocchi terra, avrai compiuto la tua missione.
Il gioco non è difficile da controllare, ma fa sfoggio di una buona grafica e la sequenza di chiusura è caratterizzata da un effetto sonoro particolarmente minaccioso e rimbombante che è udibile quando il video si riempie di aeroplani simili al tuo a cui hai aperto la strada per lo sbarco.

```
10 REM VOGON
20 GOSUB1000
30 GOSUB900
40 FOR MOVE=0 TO MAX STEP INC
50 S=PEEK(653)
60 IFSANDF=0THENGOSUB800
70 IFS=0ANDF=1THENF=2
100 ON F GOSUB400,500,600
150 IFFANDB>=8185THENGOSUB700
200 PRINT"■■■SCORE"SC;
210 IFF=0THENPRINTTAB(16)"■FIRE!"
300 NEXT MOVE:GOTO30
400 REM UPDATE BOMB          (F=1)
410 POKEB,32:B=B+1:P=PEEK(B)
420 IFP=30THEN700
430 POKEFNC(B),13:POKEB,27
440 Z=B-7680/22:POKES2,220+22*(Z-INT(Z))
450 V=V+A
490 RETURN
500 REM SLOWING (F=2)
510 V=V-D
520 POKEB,32
530 IFV<=0THENF=3:RETURN
540 B=B+1:IFPEEK(B)=30THENSC=SC+5:GOTO700
```

```
550 POKEFNC(B),5:POKEB,31
560 POKES2,200+VAND255
590 RETURN
600 REM DROP    (F=3)
610 POKEB,32
620 B=B+22:IFPEEK(B)=30THENSC=SC+10:GOTO700
640 POKEFNC(B),5:POKEB,33
690 RETURN
700 REM EXPLOSION
710 POKES1,0:POKES2,0:POKES3,0:POKES4,201
720 POKEFNC(B),2:POKEB,34
730 FORI=1TO100:NEXT
740 POKES4,0
750 POKEB,32
780 F=0
790 RETURN
800 REM LAUNCH
810 POKES3,220:PRINT"■"TAB(16)"      "
820 B=X+22
830 FORI=1TOFNR(3)
840 POKES3,PEEK(S3)-10
850 POKEB,32
860 B=B+22:POKEFNC(B),5:POKEB,31
870 NEXT
880 POKES3,0:F=1
890 RETURN
900 REM MOVE SHIP
910 POKEX,32
920 X=X+1
930 IFPEEK(X+1)=30THEN2000
940 POKEX,28
950 POKEFNC(X+1),13:POKEX+1,29
960 IFX=8184THEN3000
999 RETURN
1000 REM SET UP
1010 PRINT"■":POKE36879,8
1020 POKE36878,44
1030 POKE51,0:POKE52,28:POKE55,0:POKE56,28
1040 FORI=27*8TO7+8*(27+7)
1050 READA:POKEI+7168,A:NEXT
1100 DATA32,32,40,250,250,40,32,32
1110 DATA0,3,10,43,191,191,42,10
1120 DATA0,192,160,232,254,254,168,160
1130 DATA126,126,66,66,126,126,66,66
1140 DATA48,48,60,63,63,60,48,48
1150 DATA0,0,0,0,0,0,0,0
1160 DATA0,0,126,126,60,60,24,24
1170 DATA0,68,16,56,16,0,68,0
1300 REM SHIP DATA
1310 A=1:D=2:REM ACCELERATION & DE...
1320 X=7701
1330 MAX=1:INC=1:REMMAX&INCREMENT MOVE
1350 REM CITY BIT
1360 CH=0: REM HEIGHT
1370 CW=10:REM WIDTH
1380 CC=8175:REM CENTRE
1400 FORI=0TOCW
1410 P1=CC-I:P2=CC+I
1420 FORJ=0TOCH-I
1430 POKEP1-22*J,30:POKEP2-22*J,30
```

```
1440 NEXTJ,I
1500 DEF FNC(X)=X+30720
1510 DEF FNR(X)=INT(RND(1)*X)
1600 S1=36874:S2=S1+1:S3=S2+1:S4=S3+1
1700 POKE36869,255
1999 RETURN
2000 REM  DEAD
2010 POKES1,200:POKES2,0:POKES3,0:POKES4,0
2020 POKEX,32:POKEX+1,32
2030 X=X+22:IFX>=8185THEN2070
2040 POKEFNC(X),13:POKEX,29
2050 POKEFNC(X+1),13:POKEX+1,29
2060 GOTO2020
2070 FORI=1TO50
2080 FORJ=-1TO1
2090 POKE36864,PEEK(36864)+J:POKE36865,PEEK(36865)-J
2100 NEXTJ,I
2110 POKES1,0
2120 PRINT"[illegible]YOU FAILED!"
2130 FORI=1TO200:NEXT
2140 PRINT"[illegible]HIT A KEY TO TRY AGAIN"
2150 POKE198,0:WAIT198,1:POKE198,0
2160 RUN
3000 REM INVASION!
3010 POKES1,135:POKES2,195:POKES3,136
3020 POKES4+1,111
3030 FORI=1TO500
3040 X=FNR(253)*2+7680
3050 POKEFNC(X),13:POKEX,29
3060 POKEFNC(X+1),13:POKEX+1,29
3070 NEXT
3080 PRINT"[illegible]YOU SUCCEEDED!"
3090 PRINT"[illegible]SCORING":PRINT"[illegible]"SC
3100 PRINT"[illegible]WANT TO DESTROY"
3110 PRINT"[illegible]ANOTHER HELPLESS CITY?"
3120 PRINT"[illegible]HIT A KEY"
3130 POKE198,0:WAIT198,1:RUN

READY.
```

Come scrivere programmi migliori

di Tim Hartnell

Questo libro contiene molti, ottimi programmi, come del resto la maggior parte delle riviste specializzate in computer. D'altra parte, per quanto siano buoni i programmi proposti dalle pubblicazioni, è certo che far "girare" un programma parzialmente e completamente scritto da te ti offre un piacere maggiore. Variando i programmi, dai il tuo tocco personale, rifletti i tuoi desideri, la tua creatività. Questo è un modo eccellente per migliorare i programmi e alla fine, naturalmente, diverrai un programmatore migliore e la tua immaginazione sarà potenziata.
I programmi, presentati dai periodici o da libri come questo, sono un ideale trampolino di lancio per i tuoi progressi in questo settore. Scoprirai che perfino la pubblicità sul software può ispirarti delle idee. Devi solo leggere la descrizione dei programmi commercialmente disponibili e farai il primo passo verso la creazione del "tuo" programma. Naturalmente, non ti è consentito di violare la legge sul diritto d'autore per quanto riguarda i dati e le informazioni illustrati sul video, il nome del programma, o i nomi dei "caratteri" all'interno del programma. Probabilmente scoprirai che il programma, a mano a mano che si sviluppa, prenderà vita propria, crescerà e si evolverà staccandosi dallo schema originale finché il tuo gioco risulterà concepito e realizzato in un modo completamente nuovo.

Qualsiasi cosa tu faccia, non far passare mai come tuo il lavoro di un altro. Adatta e migliora i programmi pubblicati con ogni mezzo, ma poi non presentarli ai periodici come se fossero originali. Ho perduto il conto delle volte in cui mi è stata proposta la pubblicazione di un programma che era stato tratto da uno dei miei libri. Ricordati che i libri e i periodici specializzati in giochi e computer sono tutti ottimi spunti per le tue idee. Troverai sulla tua strada sempre nuove idee per giochi, forme di caratteri, suoni, conclusioni a sfondo drammatico, e così via. È opportuno che tu ne prenda nota. In questo modo non ti troverai mai a corto di idee e sarai in grado di unire il materiale per produrre programmi migliori e capaci di tener desta l'attenzione del giocatore più a lungo.

I giochi si dividono in tre categorie. È utile quindi che tu individui la categoria alla quale appartiene il programma che proponi **prima** di iniziarlo, dato che la categoria del gioco altera materialmente il modo di dare inizio alla programmazione. Ciò non significa che a mano a mano che lo sviluppi, il programma non possa passare da una categoria all'altra, né che un particolare gioco non possa svilupparsi tra due categorie, ma è tuttavia utile considerare i vari gruppi separati per avere idee chiare. Le tre categorie sono:

1) Board games (scacchi, dama, e simili)
2) Arcade games (giochi di movimento veloce, di fantasia, pieni di rumori, in tempo reale)
3) Giochi di fortuna (come Roulette e Snap)

Nei "Board games" la qualità del gioco è più importante della velocità della risposta. Negli "Arcade games" il movimento deve essere mantenuto ad ogni costo, anche se talune "intelligenze" tra i tuoi invasori Marziani devono essere, a tale scopo, sacrificate. I giochi di fortuna si basano sulla facilità di gioco e sulla casualità vera e propria in modo maggiore rispetto alle altre due categorie.

Scoprirai che i programmi dei giochi si dividono in generi, i quali a loro volta sono suddivisioni delle tre categorie di cui sopra. Molti "board" games sono varianti della dama o degli scacchi, molti "arcade" games erano originariamente giochi del tipo "Invasori dello spazio" mentre i giochi di fortuna sono nati nel "mondo reale" dei dati e delle carte. Prestare attenzione alla descrizione di un programma o a una macchina da gioco, per provare ad indovinare la categoria a cui appartiene il gioco che stai osservando, può aiu-

tarti e ispirarti nuove idee che si adattino a quel particolare genere di giochi. Esiste un concetto nell'ambito della programmazione — chiamato generalmente "programmazione strutturata" — secondo il quale la disciplina è essenziale all'inizio del processo di scrittura dei programmi. Pur essendo meno interessante del fatto di sedersi subito davanti al computer, alla fine il programma prodotto sarà migliore.
Una volta scrissi un programma chiamato Dome Dweller (Abitante di cupole) un programma a simulazione nel quale il giocatore ha il controllo di una "cupola lunare" e deve decidere quali prodotti produrre e vendere al fine di comprare ossigeno e cibo per gli abitanti del posto (vedi *The book of Listings,* scritto da Jeremy Ruston e edito dalla BBC). Dopo aver deciso lo schema generale da adottare, eleborai i dati e le informazioni da scrivere sul video, nel modo seguente:

Le scorte di ossigeno sono scarse
All'interno della tua cupola vivono 96 persone nell'anno 3
Il credito in denaro è di $ 5.693
Le spese di manutenzione annuali ammontano a $ 226
Le cisterne di ossigeno sono di 811 unità
L'ossigeno costa $ 8 per unità
Ogni abitante di cupola necessita di 5 unità all'anno
Le riserve di cibo ammontano a 2122
Ogni abitante necessita di 3 unità all'anno ($ 6 ognuno, $ 576 per la cupola. Ciò basterà per 7 anni alla popolazione attuale)
Tu puoi commerciare le tue straordinarie sculture lunari con la gente che vive nelle altre cupole. Consumi 2 unità di ossigeno per la costruzione di un'unica scultura, e le vendi a $ 30.

Come hai probabilmente indovinato da questo "printout" l'idea del programma è quella di decidere quante "straordinarie sculture lunari" devi costruire e vendere al fine di comprare ossigeno e cibo e pagare il "mantenimento annuale". Il problema in questo particolare programma è che per costruire ogni scultura si consuma ossigeno cosicché il giocatore si trova combattuto fra il desiderio di guadagnare e la necessità di usare l'ossigeno intelligentemente.
Puoi provare a scrivere un programma come questo da solo. In tal caso potresti riuscire a scrivere un programma divertente e ciò svilupperebbe notevolmente la tua capacità

di programmazione. La prima cosa da fare è un elenco di ciò che il programma deve compiere:

Stabilire le variabili necessarie
Comunicare al giocatore in quali condizioni si trova la cupola
Chiedere quanto ossigeno bisogna comprare
Controllare se si ha il denaro sufficiente. In caso positivo comprarlo, altrimenti ritornare e chiedere di nuovo
Chiedere quanto cibo deve essere comprato
Controllare se il denaro è sufficiente. In caso positivo comprarlo, altrimenti ritornare e chiedere di nuovo
Rinnovare la quantità di ossigeno
Rinnovare la quantità di cibo
Ridurre il totale del denaro lasciato
Domandare quante sculture devono essere costruite
Controllare se c'è l'ossigeno sufficiente per la costruzione di quel numero di statue. In caso negativo ritornare indietro e chiedere di nuovo
Ridurre la quantità di ossigeno per l'ammontare necessario alla costruzione del numero di sculture specificato. Incrementare il totale del denaro per considerare il prezzo delle sculture costruite
Incrementare leggermente l'intera popolazione, aggiungere uno all'"anno attuale"
Controllare se le riserve di cibo sono sufficienti per alimentare l'intera popolazione
Verificare se l'ossigeno è sufficiente per l'intera popolazione
Verificare se c'è un po' di denaro
Se qualcuna fra queste condizioni è negativa (per esempio, il cibo non è sufficiente) trasmettere l'azione ad un sottoprogramma "fine-gioco"
Se tutte le condizioni sono positive, ripetere di nuovo le istruzioni per comunicare al giocatore le condizioni in cui si trova la cupola, e continuare a ripetere le istruzioni.

Probabilmente saresti capace di scrivere un programma "Dome Dweller" usando le voci qui sopra elencate e l'informazione campione "printout". C'è comunque un segreto che vorrei svelarti e che risolve quasi istantaneamente i problemi legati alla programmazione. In realtà, sei in grado

di scrivere tutte le parti vitali di un programma come questo quando ne hai eleborato soltanto un rozzo schema generale, prima quindi di passare ai dettagli. E una volta che hai la struttura, puoi elaborarla per tutto il tempo che vuoi, sapendo che, mentre fai ciò, il programma che stai sviluppando è in ogni momento operante. Non devi aspettare la fine per metterlo in funzione e vedere come tu stesso ti comporti. Il "segreto" è di mantenere l'intero programma all'interno di una serie di sottoprogrammi a disposizione, ognuno dei quali mantenuto in un circuito di istruzioni continuo. Ecco come ciò che ho riferito può adattarsi a questo programma. Le primissime righe che inserirai nel tuo computer saranno le seguenti:

```
10 REM DOME DWELLER
20 GOSUB 1000: REM ASSEGNAZIONE DELLE VA-
RIABILI
30 GOSUB 2000: REM VISUALIZZAZIONE DELLA SI-
TUAZIONE DELLA CUPOLA
40 GOSUB 3000: REM OSSIGENO
50 GOSUB 4000: REM CIBO
60 GOSUB 5000: REM SCULTURA
70 GOSUB 6000: REM AUMENTO DELLA POPOLA-
ZIONE
80 GOSUB 7000: REM CONTROLLO DELLA SITUA-
ZIONE DELLA CUPOLA
90 IF (tutte le condizioni sono positive, da GOSUB
7000) THEN GOTO 30
100 REM Fine del gioco...
```

Come tu stesso puoi vedere, una volta che hai stabilito il "loop principale delle istruzioni", è relativamente facile compilare uno ad uno i sottoprogrammi, e allo stesso tempo sperimentarli ed elaborarli. In questo modo otterrai un ottimo programma. La sola cosa che adesso ti serve è un elenco delle variabili che userai insieme al programma. Sono dell'opinione che il modo migliore per far ciò sia quello di utilizzare nomi espliciti per le variabili. In questo modo, mentre stai programmando, non devi perder tempo nel verificare, per esempio, se AA indica la popolazione o il numero delle unità di ossigeno consumate per costruire ognuna delle sculture. Per facilitare il più possibile i programmi, nella trasmissione di questi fra differenti computer, puoi re-

star fedele ai nomi variabili a due lettere, o puoi usufruire (se il tuo computer lo permette) di nomi lunghi (come OXYUSE per la quantità di ossigeno usata) per le variabili. Di conseguenza non avrai dubbi sul significato di ogni nome di variabile. Per dimostrarti l'efficacia di questo accorgimento e per illustrarti un ulteriore vantaggio dei nomi espliciti per le variabili, ecco le variabili usate nel Dome Dweller:

FOLK — la popolazione della cupola
CASH — denaro in cassa
FOOD — le riserve di cibo a disposizione
FOODCOST — quanto costa ogni unità di cibo
FOODNEED — quante unità di cibo sono consumate da ogni persona nell'arco di un anno
ARTCOST — quanto ossigeno è consumato per costruire ogni pezzo di scultura
ARTPAY — a quanti dollari ogni pezzo è venduto
OXY — le riserve di ossigeno a disposizione
OXYNEED — quante unità di ossigeno sono consumate da ogni persona nell'arco di un anno
OXYCOST — quanto costa ogni unità di ossigeno che deve essere comprata
REPAIR — il costo annuale delle riparazioni concernenti la cupola
YEAR — l'anno di vita della cupola

Sebbene i nomi espliciti di variabile consumino maggiore memoria rispetto ai nomi di variabile a una o a due lettere, il loro uso permette di seguire un programma in modo molto facile, controllando ciò che compie in realtà ogni sezione del programma. Inoltre, e questo è l'ulteriore vantaggio che ho menzionato, al momento di scrivere il programma, è oltremodo facile inserire le formule richieste per operazioni di calcolo. Con questo voglio dire che se, per esempio, desideri includere (come faccio io in questo programma) un'indicazione di quanto ossigeno è necessario ogni anno, moltiplichi semplicemente il numero delle persone presenti nella cupola (FOLK) per il numero di unità di ossigeno di cui ogni persona necessita in un anno (OXYNEED). A questo punto puoi includere entro le "print" le condizioni riguardanti la cupola come, ad esempio:

```
PRINT "CI SONO"; FOLK; "NELLA CUPOLA"
PRINT "NELL'ANNO"; YEAR
PRINT "OGNI PERSONA ABBISOGNA DI"; OXY-
      NEED; "UNITÀ DI"
PRINT "OSSIGENO OGNI ANNO,"; OXYNEED * FOLK;
"FABBISOGNO ANNUALE DI OSSIGENO"
PRINT "PER L'INTERA CUPOLA"
```

In questo modo è anche più facile verificare se è possibile comprare. Per esempio, per comprare cibo, potresti dire:

```
PRINT "QUANTO CIBO VUOI COMPRARE?"
INPUT A
IF A * FOODCOST>CASH THEN GOTO (un'altra A)
```

I consigli che seguono possono migliorare i tuoi programmi facendo uso di una "programmazione strutturata":

— redigere un campione di "printout" o un modello dimostrativo dei dati e delle informazioni illustrati sul video

— redigere una lista di ciò che il programma deve compiere ogni volta che attraversa il "loop principale di istruzioni"

— trasformare questa lista in una serie di sottoprogrammi a disposizione

— se possibile, usare espliciti nomi per le variabili

Se stai progettando dei programmi che poi saranno usati da altri, assicurati che sia estremamente chiaro ciò che il giocatore dovrebbe fare mentre conduce il programma. Includere una lunga serie di istruzioni non è di molta utilità, specialmente quando la memoria è limitata, ma è in ogni caso opportuno prenderne nota. Inoltre, i suggerimenti destinati all'utente dovrebbero essere espliciti (come INSERISCI IL NUMERO DELLE MOSSE CHE VUOI EFFETTUARE) e dovrebbero avvertire riguardo i limiti che saranno posti sull'input (CON QUANTE CARTE VUOI INCOMINCIARE: 1, 2, o 3 ?, per esempio).
Non puoi supporre che sarai sempre presente ogniqualvolta un programma è in funzione, quindi, dovresti fare del tuo meglio per renderne il funzionamento il più possibile facile e sicuro. Se è possibile, aggiungi dei sottoprogrammi che eliminino gli errori.

In tal modo se si è precedentemente inserito un sottoprogramma alternativo e c'è stato un errore, questo non rovinerà il programma e non lo renderà incapace di dare buoni risultati in seguito.
Se leggi questa parte del libro da cima a fondo più di una volta e tenterai di applicare le idee qui proposte al tuo lavoro di programmazione, la qualità del tuo lavoro migliorerà in modo significativo. Ciò ti permetterà anche di dedicare più tempo al miglioramento e al perfezionamento di un programma e minor tempo al semplice lavoro meccanico di renderlo operativo.

GLOSSARIO

A

Accumulatore — Il posto all'interno del computer entro il quale si compiono calcoli aritmetici e dove i risultati di tali calcoli sono immagazzinati.
Algoritmo — La serie di passi compiuti dal computer per risolvere un particolare problema.
Alfanumerico — Questo termine è generalmente usato in relazione ad una tastiera. Esempio: "è una tastiera alfanumerica". Ciò significa che la tastiera ha sia lettere sia numeri. Indica anche "la serie di caratteri" del computer e comprende i numeri e le lettere che il computer può riprodurre sul video.
ALU (Arithmetic/Logic Unit) — La parte del computer adibita ad operazioni aritmetiche (come l'addizione e la sottrazione) e in cui vengono prese le decisioni.
AND — Un'operazione logica "booleana" che il computer usa nei suoi processi decisionali. È basata sull'algebra di "Boole", un sistema sviluppato dal matematico George Boole (1815-64). Nell'algebra di Boole le variabili di un'espressione sono soggette alle operazioni logiche come AND, OR, NOR, ecc.
ASCII — Sta per American Standard Code for Information Exchange (Codice Americano Standard per lo Scambio di Informazioni) ed è il sistema in codice più largamente usato per la lingua inglese con caratteri alfanumerici. Ci sono 128 lettere maiuscole e minuscole, cifre e alcuni caratteri speciali. Il codice ASCII trasforma i simboli e le istruzioni di controllo in combinazioni binarie a sette "bit".
ASSEMBLER — Un programma che trasforma i programmi scritti nel linguaggio Assembly in codice macchina (che il computer può comprendere in modo diretto). Il linguaggio Assembler è un linguaggio di programmazione a basso livello che usa le combinazioni facilmente memorizzabili di due o tre lettere per rappresentare una particolare istruzione che l'Assembler poi trasforma affinché il computer possa comprenderlo. Esempio: ADD (aggiungere) e SUB (sottrarre). Un computer programmatico nel linguaggio Assem-

bler tende a lavorare più velocemente di un altro programmato in un linguaggio di alto livello come il BASIC.

B

BASIC —È un acronimo per Beginners All-Purpose Symbolic Instruction Code. È il linguaggio per computer più largamente usato nel settore dei micro-computer. Sebbene sia stato molto criticato, ha il vantaggio di essere molto facile da imparare. Molte espressioni BASIC sono simili all'inglese comune.
BAUD — Viene da Baudot, un pioniere della comunicazione telegrafica. Baud misura il livello di trasmissione delle informazioni ed è approssimativamente uguale a un bit per secondo.
BCD — Un'abbreviazione per Binary Coded Decimal (Binario Codificato Decimale).
Benchmark — Un test con cui possono essere misurate certe funzioni del computer. Ci sono molti "test standard Benchmark", che generalmente sperimentano soltanto la velocità. Questo è un aspetto raramente importante in un microcomputer, il tipo di computer piú interessante per un potenziale compratore.
Binario — Un sistema numerico che usa soltanto i numeri zero e uno.
Bit — Un'abbreviazione per Binary Digit (cifra binaria). È la piú piccola unità di informazione che un circuito di computer può riconoscere.
Boole, Algebra di— Il sistema algebrico sviluppato dal matematico George Boole che usa numerazioni algebriche per esprimere relazioni logiche (vedi AND).
Bootstrap — Un breve programma o sottoprogramma che viene letto all'interno del computer quando è acceso per la prima volta. Orienta il computer ad accettare i programmi successivi, di lunghezza maggiore.
Bug — Un errore che impedisce il funzionamento del programma. Sebbene sia generalmente usato per indicare solo un difetto o un errore nel programma, il termine bug può anche essere usato per un difetto nell'hardware del computer.

Bus — Un numero di conduttori usati per trasmettere segnali, dati e istruzioni.
Byts — Un gruppo di cifre binarie che compone una parola di computer. Generalmente i bits presenti in un byte sono otto.

C

CAI — Computer Assisted Instruction
CAL — Computer Assisted Learning. Il termine è generalmente usato per descrivere quei programmi che coinvolgono lo studente in processi di apprendimento.
CHIP — Il termine che si usa per indicare l'intero circuito, che è inciso su un piccolo pezzo di silicio. Il chip è, naturalmente, il cuore del computer.
CLOCK — Il congegno di sincronizzazione all'interno del computer che sincronizza le sue operazioni.
COBOL — Un linguaggio ad alto livello che deriva dalle parole Common Business Orientated Language. Il COBOL è usato principalmente per la schedatura e il mantenimento di ciò che è stato registrato.
Comparator — Un congegno che mette a confronto due cose e produce un segnale collegato con la differenza tra i due.
Compiler — Un programma che trasforma i linguaggi di programmazione ad alto livello in codici binari da macchina. In questo modo i programmi scritti in linguaggi ad alto livello possono essere utilizzati dal computer.
Complement — Un numero che è derivato da un altro secondo regole stabilite.
Computer — Un congegno con tre principali capacità o funzioni:
1) accettare i dati
2) risolvere i problemi
3) fornire risultati.
CPU — Sta per Central Processing Unit. È il cuore dell'intelligenza del computer dove si gestiscono i dati e si sviluppano le istruzioni.
Cursore — Un carattere che appare sul video quando il computer sta compiendo le sue operazioni. Esso rivela do-

ve il prossimo carattere sarà stampato. Su un computer ci sono generalmente "tasti di controllo cursore" per permettere all'utente di spostare il cursore sul video.

D

Dati — Informazioni presentate in una forma che il computer può elaborare.
Debug — Il termine che si usa quando si esamina un programma e si correggono eventuali errori, cioè, trovare e rimuovere i bugs, i difetti.
Digital Computer — Un computer che opera su quelle informazioni che si presentano come digitali.
Disk/Disc — È un disco di plastica sensibilizzato magneticamente, un poco più piccolo di un "45 giri". È usato per l'immagazzinamento dei programmi e per ottenere dati. I dischi sono notevolmente più veloci da caricare rispetto ad una cassetta della stessa lunghezza di programma. Si può accedere molto velocemente su un disco mentre un programma sta operando per ottenere ulteriori dati.
Display — L'output visivo del computer, generalmente su un video, o su uno schermo di monitor.
Dot Matrix Printer — Una stampante che stampa i listati di un programma o di ciò che appare sul video. Ogni lettera e ogni carattere sono composti da un certo numero di "dots", cioè di punti. Più alto è il numero di punti per ogni carattere e migliore sarà la qualità operativa della stampante.
Dynamic Memory — Un'unità di memoria all'interno del computer che "dimentica" ciò che contiene quando viene tolta l'alimentazione elettrica.

E

Editor — Questo termine è generalmente usato per designare quella sezione del computer che permette al programmatore di cambiare le istruzioni di un programma mentre lo sta scrivendo.

EPROM — Sta per Erasable Programmable Read-Only Memory. È come il ROM nel computer, con la sola differenza che è abbastanza facile inserire materiale all'interno di una EPROM che non sparisce quando si toglie l'alimentazione. Le EPROM devono essere esposte a forti raggi ultravioletti se si vuole cancellarle.

Error Messages (Messaggi di errore) — L'informazione data da un computer che indica dove è stato commesso un errore nella codificazione durante una parte del programma. L'informazione è trasmessa dal computer che si ferma e stampa una parola, o una parola e dei numeri, o soltanto una combinazione di numeri, in fondo al video. Questo rivela quale errore è stato fatto. Gli errori più comuni includono l'uso della lettera O invece dello zero in una linea, o l'omissione in una espressione di entrambe o di una delle parentesi, o l'errore nella definizione di una variabile.

F

File — Una serie di item d'informazione collegati fra loro e organizzati in modo sistematico.

Floppy Disk — Un disco magnetico relativamente poco costoso, usato per immagazzinare le informazioni del computer, e così chiamato perché molto flessibile (vedi Disk/Disc).

Flow Chart — Un diagramma tracciato prima di scrivere un programma nel quale le principali operazioni sono racchiuse entro rettangoli o altre forme e connesse a frecce attraverso linee per rappresentare "loop" di istruzioni, e le decisioni scritte fra parentesi. Ciò ti aiuta a scrivere un programma in modo molto più semplice perché trappole come loop infiniti o variabili non definite possono essere scoperte in ogni fase. Può risultare utile scrivere questo diagramma per programmi molto brevi, ma è senz'altro conveniente se si vuole creare un programma più lungo.

Firmware — Ci sono tre tipi di "ware" nei computer: cioè programmi software "temporanei"; hardware, come quelli contenuti nelle ROM permanenti; e firmware, nel quale l'informazione è relativamente permanente, come in una EPROM (vedi EPROM).

Flip-Flop — Un circuito che mantiene in memoria una condizione elettrica finché questa non è cambiata nella condizione opposta da un segnale.
FORTRAN — Sta per FORmula TRANslation (traduzione di formula). È un linguaggio di computer ad alto livello, orientato verso problemi matematico-scientifici.

G

Gate — Un circuito elettrico che, sebbene possa captare uno o più segnali in arrivo, manda in uscita soltanto un singolo segnale.
Graphics — Informazione grafica, in opposizione alle informazioni fornite da lettere e numeri.

H

Hard Copy — Uscita di computer il cui supporto è permanente.
Hardware — Le parti fisiche di un computer (vedi anche software e firmware).
Hexadecimal (Hex) — Un sistema numerico con base sedici, Sono usate le cifre da zero a nove e le lettere A, B, C, D, E, F per la rappresentazione dei numeri. A è uguale a 10, B è uguale a 11, C è uguale a 12, e così via. Hex è spesso usato dagli utenti di microcomputer.
Hex Pad — Una tastiera specificamente progettata per inserire numerazioni esadecimali.
High Level Language (Linguaggio ad alto livello) — Un linguaggio di programmazione che permette all'utente di parlare con il computer più o meno in lingua inglese. In generale più è alto il livello del linguaggio (cioè, più è simile all'inglese) più lungo sarà il tempo impiegato dal computer per tradurlo in un linguaggio che esso può utilizzare. Linguaggi a più basso livello sono molto più difficili per l'operatore umano ma generalmente offrono una esecuzione più veloce.

I

Input — L'informazione inserita nel computer attraverso una tastiera, un microfono, una cassetta o un disco.
Input/Output (I/O Device) — Un congegno che accetta le informazioni o le istruzioni dal mondo esterno, le trasmette al computer e, dopo l'elaborazione, le rinvia, o sotto una forma adattabile alla memorizzazione, o una forma comprensibile all'essere umano.
Instruction — Il dato che comanda una sola azione nell'elaborazione delle informazioni operate dal computer (noto anche come comando).
Integrated Circuit (Circuito Integrato) — Un completo circuito elettronico residente sulla superficie di un semiconduttore.
Interface — Il confine fra il computer e un periferico come la stampante.
Interpreter — Un programma che traduce, istruzione per istruzione, il linguaggio ad alto livello inserito da un operatore umano in un linguaggio che la macchina può capire.
Inverter (Invertitore) — Un "gate" logico che cambia nell'opposto il segnale inserito.
Interactive Routine (Sottoprogramma Interattivo)— Parte di un programma che è ripetuto più volte finché non si raggiunge una data condizione.

J

Jump Instruction — Un'istruzione che dice al computer di muoversi verso un'altra parte del programma, quando la destinazione di questo spostamento dipende dal risultato di un calcolo appena compiuto.

K

K — Questa lettera riporta la misura della memoria. La memoria è generalmente misurata in blocchi di "K". Un K contiene 1.024 bytes.

Keyword (parola-chiave) — La parola di inizio in una linea di programmazione, generalmente la prima parola dopo il numero della istruzione. Parole-chiavi sono STOP, PRINT e GOTO.

L

Language — I linguaggi di computer sono divisi in tre gruppi: i linguaggi ad alto livello, come il BASIC, che sono relativamente vicini all'inglese ed abbastanza facili da usare per l'uomo; i linguaggi a basso livello, come l'ASSEMBLER, in cui compaiono brevi frasi che hanno qualche collegamento con l'inglese (ADD per add e RET per return, ad esempio); e il codice macchina, che comunica più o meno direttamente con la macchina.
LCD — Sta per Liquid Crystal Diode. Alcuni computer come il TRS-80 Pocket Computer usano un "display" LCD.
LED — Sta per Light Emitting Diode. I numeri luminosi rossi che sono spesso usati negli orologi da polso o da muro sono composti da LED.
Logic — La formula matematica di uno studio di relazioni fra eventi.
Loop — Una serie di istruzioni all'interno di un programma che sono ripetute finché una particolare condizione viene soddisfatta.

M

Machine Language o Machine Code (Codice macchina) — Un codice che può essere capito e messo in pratica direttamente dal computer.
Magnetic Disk— vedi Disk e Floppy Disk.
Mainframe — I computer sono generalmente divisi in tre gruppi e il fatto di appartenere ad un certo gruppo dipende dalla grandezza del computer. Il computer più venduto è il microcomputer. I computer di media grandezza sono i minicomputer, e i computer giganti che qualche volta si vedono nei film di fantascienza sono computer "mainframe". Fino a 15 anni fa tali computer erano, in termini pratici, i so-

li disponibili.
Memory — Dentro un computer ci sono due tipi di memoria. La prima chiamata ROM, è la memoria che arriva già programmata sul computer e che dice al computer come prendere decisioni e come compiere operazioni aritmetiche. Questa memoria non è cancellata se si spegne il computer. Il secondo tipo è la RAM. Questa memoria mantiene il programma che è stato scritto sulla tastiera o che è trasmesso all'interno tramite una cassetta o un disco. La maggior parte dei computer "dimenticano" ciò che è in RAM quando vengono spenti.
Microprocessor — Il cuore di qualsiasi computer. Richiede le interfacce con le unità periferiche, l'alimentazione di energia e i congegni di input e output. In tal modo può operare come un microcomputer.
MODEM — Sta per Modulatore/Demodulatore. È un apparato che permette a due computer di parlare fra loro per telefono. Generalmente i computer usano un supporto nel quale è posto un ricevitore telefonico.
Monitor — Nel linguaggio dei computer ha due significati. Un significato è uno schermo simile a quello televisivo. Un monitor ha serie difficoltà ad adattarsi ai programmi televisivi e generalmente l'immagine prodotta su un monitor è migliore di quella prodotta da una comune Tv. Il secondo significato di un monitor si rapporta alla ROM. Il monitor di un computer è descritto come il complesso di informazioni incorporate nel computer all'atto dell'acquisto. Queste informazioni permettono di prendere decisioni e di compiere calcoli aritmetici.
Motherboard — Una struttura alla quale possono essere aggiunti circuiti extra. Questi circuiti spesso offrono facilitazioni che non sono incorporate nel computer, come quella di produrre suoni o di controllare una penna ottica.
MPU — Abbreviazione per Microprocessor Unit.

N

Nano-secondo — Un nano-secondo è un millimiliardesimo di secondo, l'unità di tempo con la quale si misura la velocità di un computer o di un microcircuito di memoria.
Non-Volatile Memory — La memoria che non si perde quando il computer è spento. Alcuni computer più piccoli

come il TRS-80 Pocket Computer hanno "non volatile memory". Le batterie mantengono il programma inserito per settecento ore.
Not - Un'operazione booleana che trasforma una cifra binaria nel suo opposto.
Null String — Una "stringa" che non contiene caratteri. Nel programma compare sotto forma di due doppie virgolette, senza niente fra di loro.
Numerico — Concerne i numeri quando sono opposti alle lettere (cioè alfabetico). Molte tastiere sono alfanumeriche, sono cioè provviste sia di numeri sia di lettere.

O

Octal — Un sistema numerico che usa otto come base e quindi le cifre da 0 a 7. Tale sistema non è oggi molto usato nel settore dei microcomputer. Il sistema esadecimale è più comune (vedi Hexadecimale)
Operating System — (Sistema operativo) — Il software e il firmware, generalmente previsti su una macchina che permette di far girare altri programmi.
OR — Un'operazione booleana che ritorna a 1 se uno o più input sono 1.
Oracle (Oracolo) — Un metodo di messaggi invianti test tramite un segnale di trasmissione televisiva. Un set di teletest è richiesto per decodificare i messaggi.
Output — Informazioni o dati trasmessi dal computer a congegni quali uno schermo come quello televisivo, una stampante o una cassetta. L'output generalmente consiste in un'informazione che il computer ha prodotto come risultato della elaborazione di un programma.
Overflow — Un numero troppo grande o troppo piccolo per essere elaborato dal computer.

P

Pad — Vedi Keypad

Pagina — Spesso usata per indicare la quantità di informazioni necessaria per riempire uno schermo televisivo. Così, vedendo una pagina del programma, è possibile analizzare la quantità di informazioni che appaiono sul video tutte in una volta.
PASCAL — Un linguaggio ad alto livello.
Periferico — Qualsiasi cosa che è collegata e controllata dal computer, come un'unità a disco, una stampante o un sintetizzatore vocale.
Port — Un connettore attraverso il quale le informazioni sono trasmesse o inserite nel computer.
Prestel — Il nome inglese per un sistema basato sulla trasmissione di informazioni via telefono da un computer centrale e sulla loro visualizzazione su uno schermo televisivo. Negli Stati Uniti una versione commerciale simile è nota come "The Source".
Program — Nel linguaggio computer può essere una lista di istruzioni che si inseriscono nel computer, oppure può essere un verbo, cioè "programmare un computer".
PROM — Sta per Programmable Read Only Memory. È un sistema che può essere programmato e, una volta che lo è stato, il programma è permanente (vedi anche EPROM e ROM).

R

Random Access Memory (RAM) — La zona di memoria entro il computer che può essere cambiata a comando dalla persona che usa il computer. Il contenuto della RAM è di solito perduto quando un computer è spento. La RAM è l'integrato che memorizza ciò che viene scritto e anche i risultati di calcoli in atto.
Read-Only Memory (ROM) — In contrasto alla RAM, l'informazione qui non può essere cambiata dall'utente e non va perduta quando si spegne il computer. I dati della ROM sono collocati dal produttore e dicono al computer il modo con cui deve prendere decisioni e come compiere calcoli aritmetici. La misura di capacità della ROM e RAM è data in unità K (vedi K).
Recursion — La ripetizione continua di una parte del programma.

Registro — Una specifica sezione della memoria nella quale uno o più parole di computer sono memorizzate nel corso delle operazioni.
Parola Riservata — Una parola che non può essere usata per una variabile in un programma perché il computer la leggerà in modo errato. Un esempio è la parola TO. Poiché TO ha uno specifico significato nel linguaggio dei computer, la maggior parte dei calcolatori respinge questa parola come nome per una variabile. Lo stesso vale per parole come FOR, GOTO e STOP.
Routine — Questa parola può essere usata come sinonimo di programma o può riferirsi a una specifica sezione all'interno del programma (vedi anche Subroutine).

S

Seconda Generazione — Ha due significati. Il primo si applica nei confronti dei computer che usano transistor, in opposizione alla prima generazione di computer che usavano valvole. La seconda generazione può anche indicare la seconda copia di un particolare programma. Susseguenti generazioni sono danneggiate da un disturbo crescente.
Semiconduttore — Un materiale che è generalmente un isolante elettrico, ma sotto specifiche condizioni può diventare un conduttore.
Serial — Informazione che è memorizzata o inviata in una sequenza, un bit alla volta.
Segnale — Un impulso elettrico che trasmette dati.
Silicon Valley — Il nome popolare dato a una zona in California dove si trovano molti produttori di semiconduttori.
SNOBOL — Un linguaggio ad alto livello.
Software — Il programma inserito nel computer dall'utente. Questo programma dice al computer cosa fare.
Software Compatible — Si riferisce a due differenti computer che possono accettare i programmi scritti per l'altro.
Static Memory — Un congegno di memoria non volatile che trattiene le informazioni per tutto il tempo che il computer è acceso. Tuttavia, non richiede ulteriori consumi di energia per mantenere in ordine la memoria.
Subroutine (Sottoprogramma) — Parte di un programma che è spesso inserita molte volte durante l'esecuzione del programma principale. Una subroutine finisce con un'istru-

zione che comanda di ritornare indietro alla istruzione successiva a quella che aveva inviato lo svolgimento del programma alla subroutine.

T

Teletext — Informazione trasmessa nella sezione superiore di un'immagine che appartiene a una trasmissione televisiva. Richiede una struttura speciale per essere decodificata e per riempire il video di informazioni riguardo il test. I messaggi teletext possono anche essere trasmessi tramite un cavo, per esempio, il servizio Prestel in Gran Bretagna o The Sources negli Stati Uniti.
Teletype — Un apparecchio simile ad una macchina da scrivere che può mandare, ricevere e stampare informazioni.
Terminale — Un'unità indipendente dell'unità centrale di elaborazione. Generalmente consiste di una tastiera e di un visore.
Time Sharing — Un processo attraverso il quale molti utenti possono aver accesso a un grande computer che si sposta rapidamente da un utente all'altro in sequenza, cosicché ogni utente ha l'impressione di essere il solo utente del computer.
Truth Table (Tavola della verità) — Una tavola matematica che elenca tutti i possibili risultati di un'operazione booleana. I risultati dipendono dalle varie combinazioni di input.

U

UHF — Ultra High Frequency (300-3000 MegaHertz).
Ultra Violet Erasing — La luce ultravioletta deve essere usata per cancellare le EPROM (vedi EPROM).

V

Variabile — Una lettera o combinazione di lettere e simbo-

li a cui il computer può assegnare un valore o una parola durante il funzionamento di un programma.
VDU — Abbreviazione per Visual Display Unit.
Volatile — Indica la memoria che "dimentica" le informazioni in essa contenute quando il computer è spento.

W

Word (Parola) — Un gruppo di caratteri o una serie di cifre binarie che rappresentano un'unità d'informazione e occupano una singola posizione di memoria. Il computer elabora una parola come singola istruzione.
Word-Processor (Elaboratore di testi) — Una macchina da scrivere altamente intelligente che permette a chi scrive di manipolare il testo, di spostarlo, per giustificare margini e per spostare interi paragrafi, se necessario, su un video prima di mandare l'informazione sulla stampante. Questi elaboratori hanno generalmente memorie, cosicché modelli di lettere e testi di lettere, scritti precedentemente, possono essere nuovamente stampati.

Traduzioni

NEW KID IN TOWN

12 — Nastro di dimostrazione — Un tipo allegro — guarda alcune
13 — delle capacità — del VIC
20 — Ciao! — il mio nome è VIC — e il tuo
30 — fedele servitore — Vorresti vedere alcune delle meravigliose
40 — cose che io posso fare? — Sì!!!
50 — Bene! Sei seduto comodamente?
60 — Guarda come cambio il margine — e il colore del video in
65 — 255 differenti modi
80 — Lasciami solo cambiare
90 — il mio colore originale
100 — Ora, suoniamo il piano. Premi solo i tasti seguenti
180 — Perchè non provare ancora? — oppure, perché — non proviamo a comporre musica?
181 — Il tuo motivo? Premi solo "Y" o "N"
250 — Vediamo se riesco a ricordarmi il motivo che stai cercando di suonare
320 — Non è esattamente Bach, ma sto imparando!!!
340 — Adesso, creiamo qualche effetto sonoro
480 — Ora per una variazione del gioco: Knocking down the wall
490 — O perché non usarlo per progettare il modello — del tuo prossimo tappeto da salotto?

COMBAT

770 — Il gioco è stato vinto da — il giocatore
780 — Un'altra partita? (Y/N)
860 — Lo scopo del gioco è di segnare 10 punti
870 — Ci sono due modi per segnare un punto
880 — 1) Colpendo il carro armato del tuo avversario 2) quando il tuo avversario urta una mina
890 — I missili voleranno sulle mine e si disintegreranno colpendo la parete esterna
900 — Ogni giocatore ha nove controlli — premi qualsiasi

tasto per continuare
1000 — Premi qualsiasi tasto per continuare
1020 — Per spostarti premi il tasto per la direzione che hai scelto
1030 — Per sparare premi il tasto F o ↑
1040 — Il missile spara verso la direzione nella quale ti sei spostato per l'ultima volta
1050 — Carro armato di sinistra
1060 — Carro armato di destra
1070 — Premi qualsiasi tasto per iniziare

TRAIL BLAZER

2020 — Per i tuoi spostamenti sul video usa il Joystick o
2960 — Premi qualsiasi tasto
2990 — Tastiera o Joystick? (K/J)

WUMPUS

180 — Frecce magiche
200 — C'è un pipistrello vicino
210 — Lanci una freccia?
260 — Qui non si può sparare
270 — L'hai mancato
300 — Hai ucciso un pipistrello
320 — Hai impiegato meno di 20 minuti. Sei sopravvissuto.
390 — C'è molta corrente, qui — attento, i pozzi!!!
420 — I vampiri ti hanno portato via!!!
440 — Ti hanno fatto cadere nella caverna
490 — Non puoi andare lì
560 — Sei caduto dentro un pozzo!
580 — Sei stato mangiato da un pipistrello!!!!!!
600 — Sei andato fuori tempo
620 — Non hai più frecce magiche

DIGIT—DODGE

310 — Tempo finito
320 — Un'altra partita (Y/N)?
370 — Troppi urti

SPACE DOCK

2070 —L'obiettivo del gioco è di pilotare il tuo shuttle fino al
2080 —molo di approdo rosso — deve evitare le stelle bianche ed anche la parte superiore e
2090 —inferiore del video. È sicuro lasciare un lato dello schermo e
2100 — riapparire sull'altro lato.
2110 — Premi qualsiasi tasto
2120 — Il tuo punteggio è segnato in alto a sinistra sul video
2130 — Cambia la direzione dello shuttle con
2140 — una rotazione in senso orario
2145 — una rotazione in senso antiorario
2150 — Premi qualsiasi tasto

MOONBASE

5010 — La tua missione è di proteggere la tua base lunare dall'alieno
5020 — Il bombardiere sta volteggiando in cielo
5030 — La base lunare è protetta da due schermi. Tu puoi essere di aiuto in due modi
5040 — 1) riparando lo schermo superiore spostando il tuo cannone-laser su di esso
5050 — 2) distruggendo l'alieno quando è abbastanza in basso
5060 — Premi qualsiasi tasto

5100 — Più a lungo sopravvivi più alto sarà il punteggio. Si segnano 500 punti se
5110 — colpisci l'alieno
5120 — Si perde una vita se l'alieno colpisce la base lunare o il tuo cannone
5130 — La partita si conclude se l'alieno atterra

FRUIT MACHINE

740 — Premere qualunque tasto per giocare

SPOTS FROM OUTER SPACE

32 — Macchie dallo spazio esterno
33 — Livello di gioco?
600 — Tutto OK
700 — Siete stati uccisi

CAR RACE

2 — Marce — F1 = prima: F7 = quarta — Pista larga?
380 — Avete percorso — miglia!!

REVERSI

6510 — Mossa illecita
9010 — Io non posso muovere ancora — Partite di nuovo
9730 — È pari e patta!!

9750 — Vinco io
9800 — Vinci tu
9900 — Un altro gioco?
10000 — Nessuno di noi due può muovere

APOLLO II

43 — Un buon lancio
1050 — Addio astronauta!!

SOUND EFFECTS

20 — Effetti sonori
30 — Esplosione
100 — Fischio di una bomba
200 — Note musicali
400 — Che pandemonio!!!
500 — Allarme rosso
700 — Raggio laser
900 — Polizia
1000 — Telefono
1100 — Nota inserita
1200 — Uccelli
1300 — Quatermass
1400 — Onde dell'oceano
1500 — L'ufo sta scomparendo
1600 — Atterraggio dell'ufo
1700 — Caccia all'ufo
1800 — Fischio del lupo
1900 — Pericolo!!!
2000 — Blips
3000 — Lasciare il porto
4000 — In azione

HANGMAN

12 — Dati: sei, basic, vic, cane, scatola, cima, mela, ritmo, inno, cielo, siringa, chiave, polso, tiranno
13 — Luce, iena, cianuro, aggiungere, morire, volare, comprare, gin, ming, braccio, vaso, veterinario, biscotto, zoppicamento, carbone
14 — Dati: struttura, paura, ring, sangue, show, impiccare, falco, re, orologio, pettine, astuto, bastone, blocco, picco, flack
15 — Dati: alce, velcro, inferno, baffo, camicia, calze, bicchiere, sella.

PONTOON

1512 — Complimenti!!! Hai fatto saltare il banco!!!
1515 — Mi dispiace! Non ce l'hai fatta!
2030 — Mi dispiace! Hai sballato!
2040 — Girare — Comprare — Stare — Inserisci uno di questi
2090 — Non posso fermarmi prima di segnare 15
2100 — Fermo su "T"
2110 — Non è possibile girare a meno di 11
2120 — Inserire Offerta L. — La posta L.
2140 — Soltanto alla prima offerta
3010 — Pontoon. Il banco vince
3020 — Vinci!!!
3025 — Il banco vince
3070 — Il banco vince
3090 — Il banco vince

SKI RUN

20 — Sinistra — destra
25 — Scia attraverso la porta

30 — Velocità — professionale
40 — Principiante

ZOMBIES

8004 — Premio
9005 — Tasti: A: per l'alto - S: per il basso - K: per la sinistra - L: per la destra
9006 — Spazio: stop
9007 — Battere qualsiasi tasto per incominciare
9501 — Un'altra partita? (sì/no)

MICRO MUSIC

280 — Inserire: "Nuovo ritmo"
290 — Inserire "Intervallo"
310 — Inserire "Mezzo" - Inserire "Alto" - Inserire "Rumore"
320 — Inserire "Volume" - Inserire "Durata"
350 — Inserire "Ritmo"
400 — Inserire "Basso"
410 — Come 310
420 — Come 320

JUPITER RUN

2090 — Sei morto!!!
2100 — E hai segnato il punteggio di soli
2110 — colpi sparati
2150 — Vuoi fare meglio?

VOGON

210 — Fuoco
400 — Prendere una nuova bomba (F:1)
500 — Rallentare
600 — Lanciare
900 — Spostare l'aereo
1300 — Inviare i dati
1310 — Accelerazione
1360 — Altezza
1370 — Ampiezza
2120 — L'hai mancato!!
2140 — Battere un tasto per provare ancora
3000 — Invasione!
3080 — Ce l'hai fatta!!!
3100 — Vuoi distruggere
3110 — un'altra città indifesa?
3120 — Batti un tasto

ANNOTAZIONI

Computer Games

GIOCHIAMO CON VIC 20

Tanti fantastici programmi, con la traduzione in italiano, appositamente ideati per questa collana e in grado di garantirvi ore e ore di svago istruttivo e divertente.

Tra i giochi spettacolari di questo libro:
IL PIANOFORTE DEL 2001 (trasformate il vostro VIC20 in un fantastico strumento musicale); LO SPLENDORE DEL DIAMANTE (Van Gogh non ha mai visto niente di simile!); I PIPISTRELLI (cinque magiche frecce vi possono salvare dai pipistrelli); GUASTATORE IN RETROGUARDIA (al comando di un caccia spaziale inseguito dalla flotta nemica); L'ATTRACCO NELLO SPAZIO (pilotate il vostro shuttle verso un porto sicuro); FRUIT MACHINE (il computer non è un imbroglione); MACCHIE DALLO SPAZIO ESTERNO (proteggete il vostro pianeta da invasori); INSEGUIMENTO NEGLI ABISSI (missione: distruggere i sommergibili nemici).

"GIOCHIAMO CON VIC 20" vi aiuterà moltissimo, giocando, a migliorare la vostra abilità fornendovi non solo tutte le istruzioni per inserire correttamente i programmi nel vostro computer, ma anche un utilissimo glossario dei termini essenziali e preziosi consigli e indicazioni su come modificare e migliorare i programmi del libro o realizzarne di nuovi.

Tanti fantastici giochi per tutta la famiglia!

CL 006-0117-9 ISBN 88-7605-117-1

L. 9.500 (i.i.)